



Ansibleを使用したNetApp SAP Landscape Managementの統合

NetApp solutions for SAP

NetApp
October 30, 2025

目次

Ansibleを使用したNetApp SAP Landscape Managementの統合	1
TR-4953：『NetApp SAP Landscape Management Integration using Ansible』	1
SAPシステムのクローニング、コピー、更新のシナリオ	1
システムの更新、コピー、クローニングのユースケース	2
論理的破損に対処する	3
ディザスタリカバリのテスト	4
Ansibleを使用して、NetApp SAP LaMaを統合	5
実装例	5
検証済みの構成と制限事項	6
ラボのセットアップ	6
SAP LaMa設定	7
SAP LaMaプロビジョニングワークフロー-クローンシステム	11
SAP LaMaによるワークフローのプロビジョニング解除-システムの破壊	19
SAP LaMaプロビジョニングワークフロー-コピーシステム	22
SAP LaMaプロビジョニングワークフロー-システムの更新	26
プロバイダスクリプトの設定とAnsibleのプレイブック	28
プロバイダ構成ファイルNetApp_clone.conf	29
プロバイダスクリプトnetapp_clone.sh	29
Ansible Playbook：NetApp_LaMa_CloneVolume.yml	37
Ansible Playbook：NetApp_LaMa_ServiceConfigRemoval.yml	38
Ansible Playbook：NetApp_LaMa_ClearMountConfig.yml	39
Ansibleのinventory.ymlの例	40
まとめ	41
追加情報の参照先	41
バージョン履歴	42

Ansibleを使用したNetApp SAP Landscape Managementの統合

TR-4953：『NetApp SAP Landscape Management Integration using Ansible』

SAP Landscape Management (LaMa) を使用すると、SAPシステム管理者は、SAPシステムのエンドツーエンドのクローニング、コピー、更新などのSAPシステム運用を自動化できます。

著者：Michael Schlosser、Nils Bauer、NetApp

ネットアップは、SAP LaMa対応の各種Ansibleモジュールを搭載しており、SAP LaMaのAutomation Studioを通じて、NetApp SnapshotやFlexCloneなどのテクノロジーにアクセスできます。これらのテクノロジーを使用することで、SAPシステムのクローニング、コピー、更新の処理を簡易化、高速化できます。

この統合は、ネットアップのストレージソリューションをオンプレミスで運用しているお客様や、Amazon Web Services、Microsoft Azure、Google Cloud Platformなどのパブリッククラウドプロバイダでネットアップのストレージサービスを使用しているお客様が利用できます。

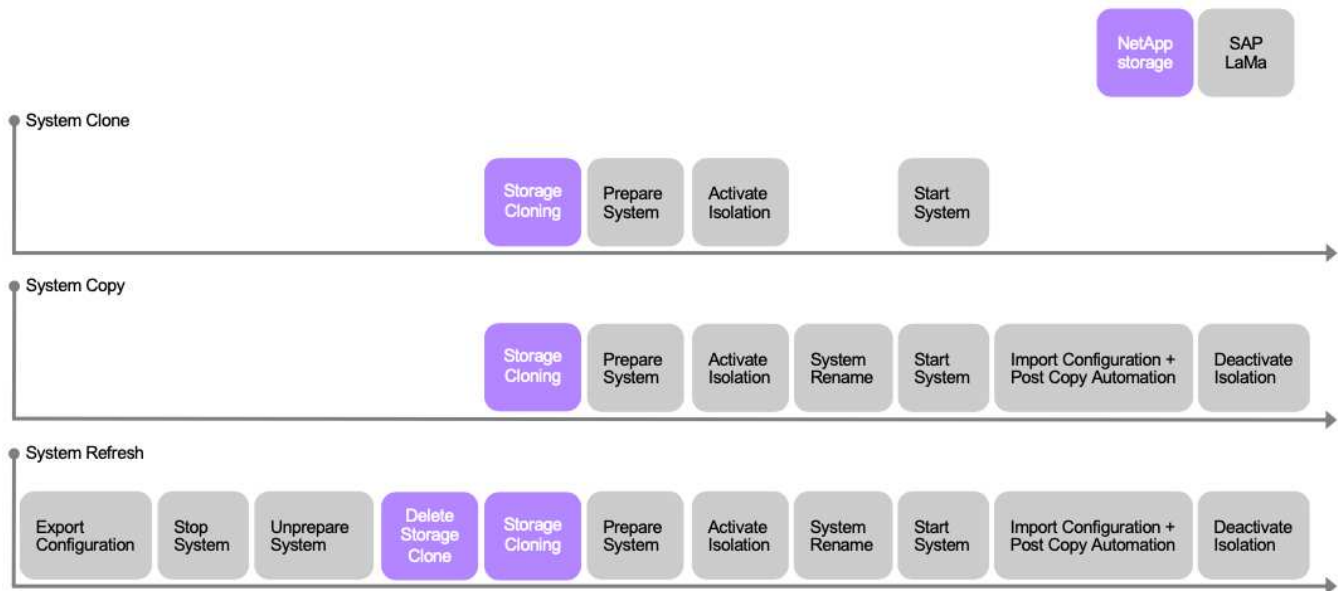
このドキュメントでは、Ansibleによる自動化を使用して、SAPシステムのコピー、クローニング、更新の処理に対応するネットアップのストレージ機能を利用したSAP LaMaの設定について説明します。

SAPシステムのクローニング、コピー、更新のシナリオ

SAPシステムのコピーという用語は、SAPシステムのクローニング、SAPシステムのコピー、SAPシステムの更新の3つのプロセスを表す同義語としてよく使用されます。ワークフローとユースケースはそれぞれ異なるため、異なる処理を区別することが重要です。

- **SAPシステムのクローン。** * SAPシステムのクローンは、ソースSAPシステムの同一クローンです。SAPシステムクローンは、一般に論理的な破損に対処したり、ディザスタリカバリのシナリオをテストしたりするのに使用されます。システムのクローニング処理では、ホスト名、インスタンス番号、およびSIDの値は変更されません。そのため、本番環境と通信できないように、ターゲットシステムの適切なネットワークフェンシングを設定することが重要です。
- **SAPシステムのコピー。** * SAPシステムのコピーとは、ソースSAPシステムのデータを含む新しいターゲットSAPシステムのセットアップを指します。たとえば、本番用システムのデータを含む追加のテストシステムを新しいターゲットシステムとして使用できます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。
- **SAPシステムの更新。** * SAPシステムの更新は、ソースSAPシステムのデータを使用した既存のターゲットSAPシステムの更新です。ターゲットシステムは、通常、品質管理システムなどのSAP転送環境の一部であり、本番用システムのデータで更新されます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。

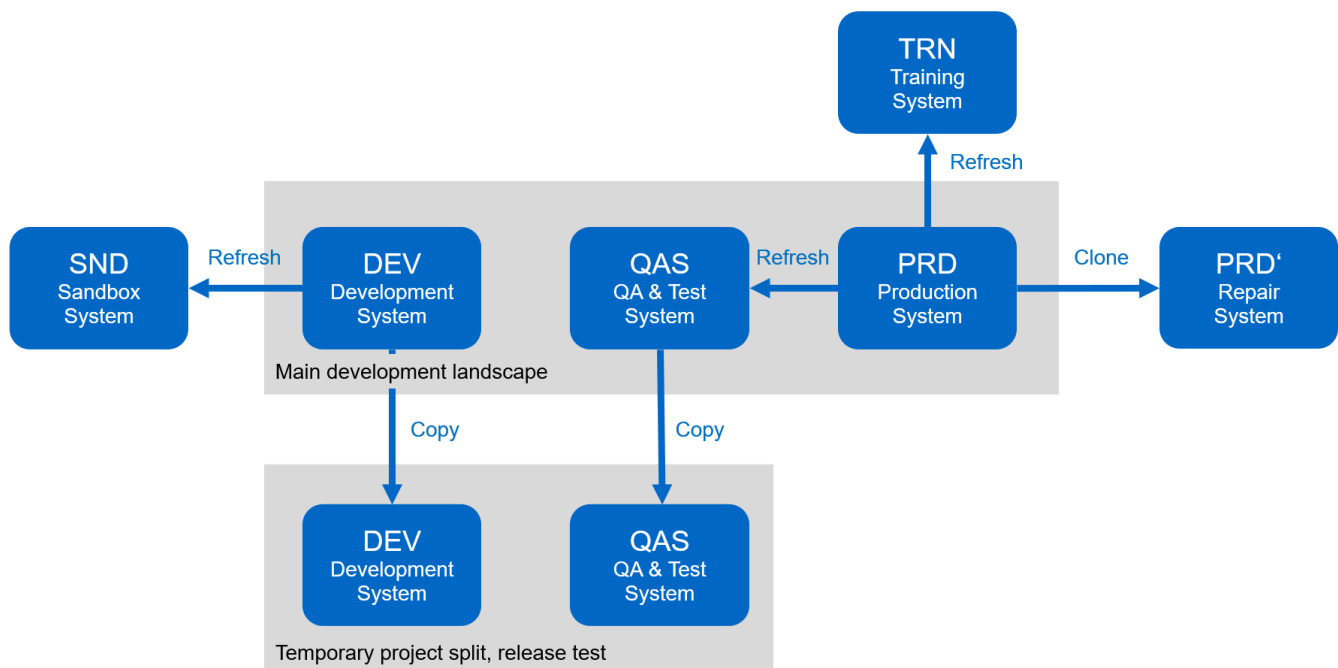
次の図に、システムのクローン、システムのコピー、またはシステムの更新の各処理で実行する必要がある主な手順を示します。紫のボックスは、ネットアップストレージ機能を統合できる手順を示しています。3つの運用はすべて、SAP LaMaを使って完全に自動化できます。



システムの更新、コピー、クローニングのユースケース

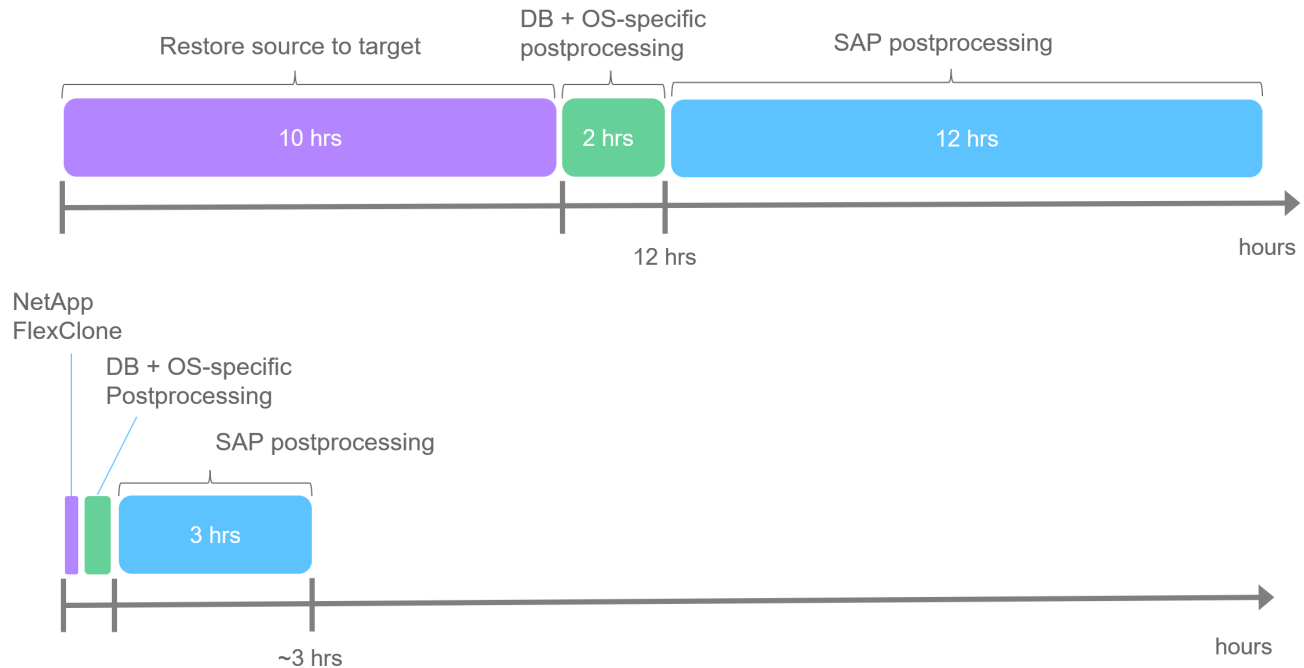
テストやトレーニングの目的で、ソースシステムのデータをターゲットシステムで使用する必要があるシナリオは複数あります。テストおよびトレーニング用のシステムは、ソースシステムのデータで定期的に更新し、現在のデータセットでテストとトレーニングが実行されていることを確認する必要があります。

このシステム更新処理は、インフラ、データベース、アプリケーションの各レイヤ上で実行される複数のタスクで構成されます。自動化のレベルによっては、数日かかる場合があります。



SAP LaMaとネットアップのクローニングワークフローを使って、インフラレイヤとデータベースレイヤで必

要なタスクを高速化し、自動化できます。SAP LaMaは、バックアップをソースシステムからターゲットシステムにリストアする代わりに、NetApp SnapshotコピーとNetApp FlexCloneテクノロジーを使用して、起動したHANAデータベースまでの必要なタスクを、次の図に示すように数時間ではなく数分で実行します。クローニングプロセスに要する時間はデータベースのサイズに左右されないため、非常に大規模なシステムでも数分で作成できます。また、オペレーティングシステムとデータベースレイヤ、およびSAPの後処理側でタスクを自動化することにより、ランタイムをさらに削減できます。



論理的破損に対処する

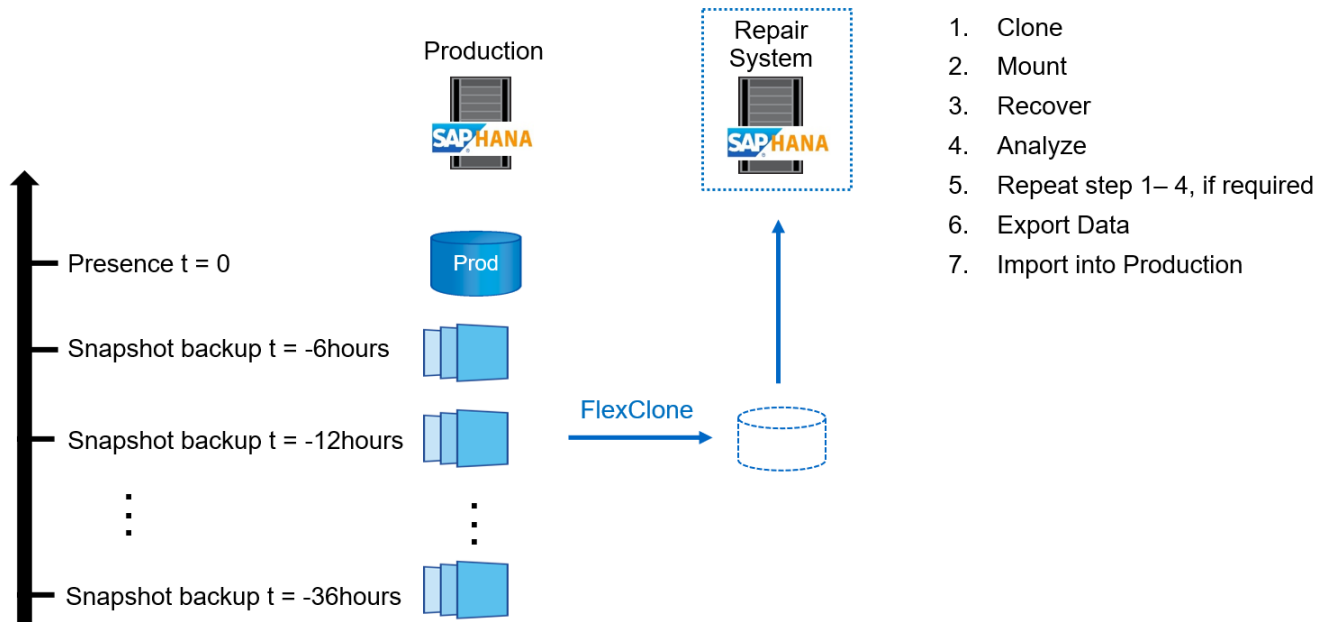
論理的破損は、ソフトウェアエラー、人為的エラー、破壊行為などが原因で発生する可能性があります。残念ながら、論理的破損は、標準的な高可用性ソリューションやディザスタリカバリソリューションでは対処できないことがよくあります。その結果、論理的な破損が発生したレイヤ、アプリケーション、ファイルシステム、またはストレージによっては、ダウンタイムを最小限に抑え、データ損失要件を許容できる範囲で満たすことができない場合があります。

最悪のケースは、SAPアプリケーションが論理的に破損した場合です。SAPアプリケーションは多くの場合、異なるアプリケーションが相互に通信してデータを交換する環境で動作します。このため、論理的な破損が発生したSAPシステムはリストアとリカバリを実行しないことを推奨します。破損が発生する前の時点でシステムをリストアすると、データが失われます。また、SAP ランドスケープは同期されず、さらにポストプロセスが必要になります。

SAPシステムをリストアする代わりに、別の修復システムで問題を分析して、システム内の論理エラーを修正する方法を推奨します。ルート原因分析には、ビジネスプロセスやアプリケーション所有者の関与が必要です。このシナリオでは、論理的破損が発生する前に格納されたデータに基づいて、修復システム（本番用システムのクローン）を作成します。リペアシステム内では、必要なデータをエクスポートし、本番システムにインポートできます。このアプローチでは、本番用システムを停止する必要はなく、最良のシナリオでは、データの損失だけでなく、ごくわずかなデータの損失も発生します。

リペアシステムを設定する際には、柔軟性とスピードが不可欠です。ネットアップのストレージベースのSnapshotバックアップでは、複数の整合性のあるデータベースイメージを使用し、NetApp FlexCloneテク

ノログを使用して本番用システムのクローンを作成できます。ファイルベースのバックアップからリダイレクトされたリストアを使用して修復システムを設定する場合、FlexCloneボリュームは数時間ではなく数秒で作成できます。

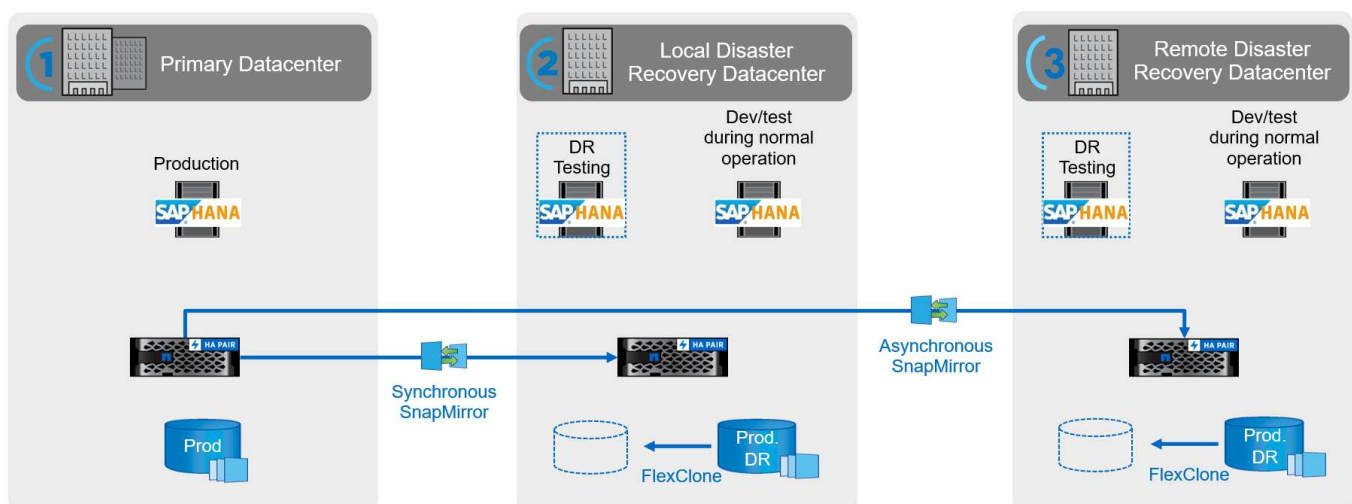


ディザスタリカバリのテスト

効果的なディザスタリカバリ戦略を策定するには、必要なワークフローをテストする必要があります。テストでは、戦略が機能するかどうか、および内部ドキュメントで十分かどうかを検証します。また、管理者は必要な手順をトレーニングできます。

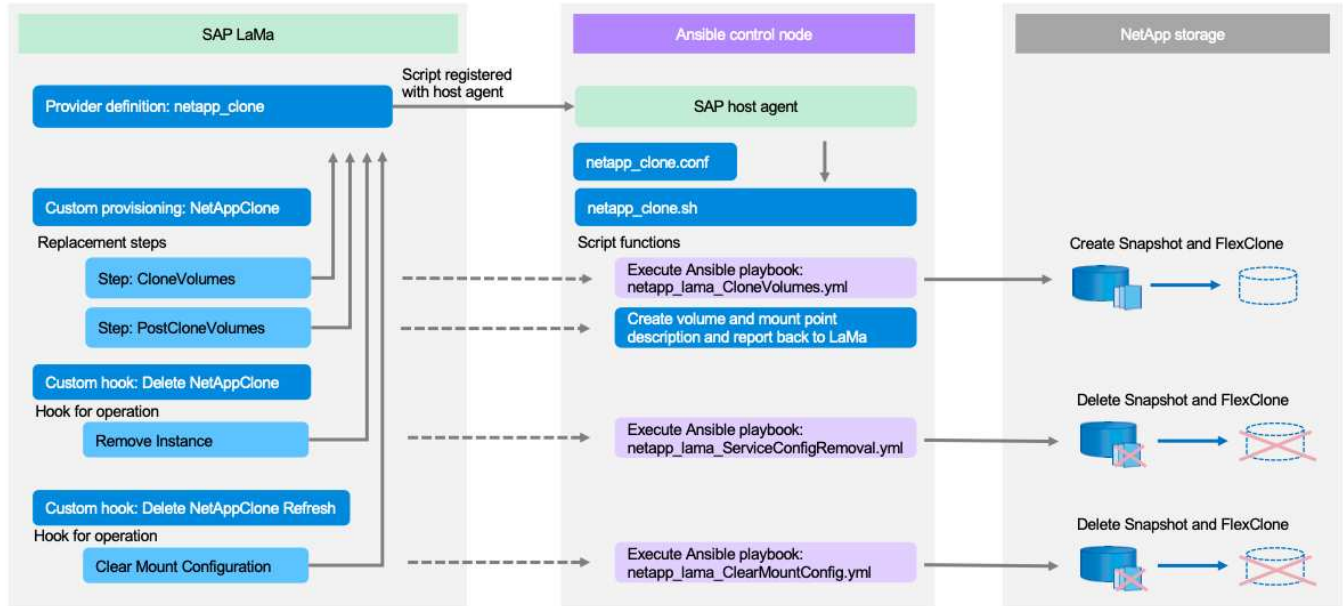
SnapMirrorを使用したストレージレプリケーションでは、RTOとRPOをリスクにさらすことなく、ディザスタリカバリのテストを実行できます。ディザスタリカバリテストは、データレプリケーションを中断することなく実行できます。非同期SnapMirrorと同期SnapMirrorのディザスタリカバリテストでは、ディザスタリカバリターゲットでSnapshotバックアップとFlexCloneボリュームを使用します。

SAP LaMaは、テスト手順全体のオーケストレーションに使用でき、ネットワークの遮断やホストのメンテナンスなどにも対応します。



Ansibleを使用して、NetApp SAP LaMaを統合

この統合アプローチでは、SAP LaMaのカスタムプロビジョニングと運用のフックを、ネットアップストレージ管理のAnsibleプレイブックと組み合わせて使用します。次の図は、LaMa側の設定の概要と、実装例の対応するコンポーネントを示しています。



Ansible制御ノードとして機能する中央ホストは、SAP LaMaからの要求を実行し、Ansibleプレイブックを使用してネットアップストレージの運用を開始するために使用されます。ホストをSAP LaMaへの通信ゲートウェイとして使用できるように、SAPホストエージェントコンポーネントをこのホストにインストールする必要があります。

LaMa Automation Studioでは、AnsibleホストのSAPホストエージェントに登録されたプロバイダが定義されます。ホストエージェント構成ファイルは、要求された処理に応じて、一連のコマンドラインパラメータを使用してSAP LaMaによって呼び出されるシェルスクリプトを指します。

LaMa Automation Studioでは、カスタムプロビジョニングとカスタムフックを定義して、プロビジョニング時にストレージのクローニング処理を実行し、システムがプロビジョニング解除されたときにクリーンアップ処理を実行します。その後、Ansible制御ノード上のシェルスクリプトによって、対応するAnsibleプレイブックが実行されます。これにより、SnapshotとFlexCloneの処理およびプロビジョニング解除ワークフローを使用したクローンの削除がトリガーされます。

NetApp AnsibleモジュールとLaMaプロバイダの定義の詳細については、以下をご覧ください。

- ["NetApp Ansibleモジュール"](#)
- ["SAP LaMaのドキュメント-プロバイダ定義"](#)

実装例

システムとストレージのセットアップでは多数のオプションを使用できるため、システムの個々のセットアップと設定の要件をテンプレートとして使用する必要があります。



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトの最新バージョンは、mailto:ng-sapcc@netapp.com [ng-sapcc@netapp.com]にEメールでリクエストできます。

検証済みの構成と制限事項

以下の原則はサンプル導入に適用されており、お客様のニーズに合わせて調整する必要がある場合があります。

- マネージドSAPシステムは、NFSを使用してネットアップストレージボリュームにアクセスし、アダプティブデザインの原則に基づいてセットアップしました。
- NetApp Ansibleモジュール（ZAPIとREST API）でサポートされているすべてのONTAP リリースを使用できます。
- 1つのネットアップクラスタとSVMのクレデンシャルが、プロバイダスクリプトの変数としてハードコードされました。
- ストレージクローニングは、ソースSAPシステムで使用されていたのと同じストレージシステムで実行しました。
- ターゲットのSAPシステムのストレージボリュームの名前は、付録のソースと同じです。
- セカンダリストレージ（SV / SM）でのクローニングは実装されていません。
- FlexCloneスプリットを実装できませんでした。
- インスタンス番号は、ソースとターゲットのSAPシステムで同じです。

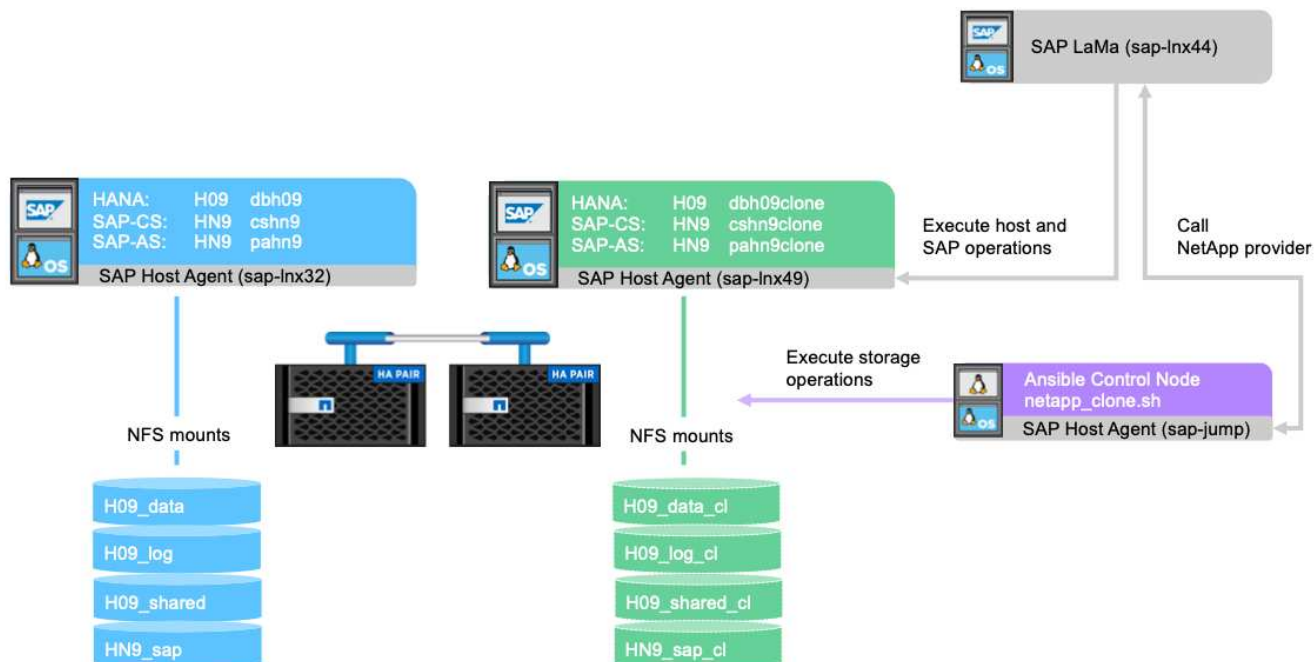
ラボのセットアップ

次の図は、ラボのセットアップを示しています。システムのクローニングに使用されるソースのSAPシステムHN9では、データベースH09、SAP CS、およびSAP ASサービスが、がインストールされた同じホスト（sap-lnx32）で実行されています "[アダプティブデザイン](#)" 有効。に従ってAnsibleコントロールノードが準備されました "[NetApp ONTAP 向けのAnsibleプレイブック](#)" ドキュメント

SAPホストエージェントもこのホストにインストールされています。NetAppプロバイダスクリプトとAnsibleプレイブックは、Ansibleコントロールノードで設定しました（を参照）。"[付録：プロバイダスクリプトの設定](#)"

ホスト sap-lnx49 SAP LaMaクローニング運用のターゲットとして使用し、隔離に対応した機能を設定しました。

システムのコピーおよび更新には、さまざまなSAPシステム（ソースとしてHNA、ターゲットとしてHN2）が使用されました。これは、Post Copy Automation（PCA）が有効になっているためです。



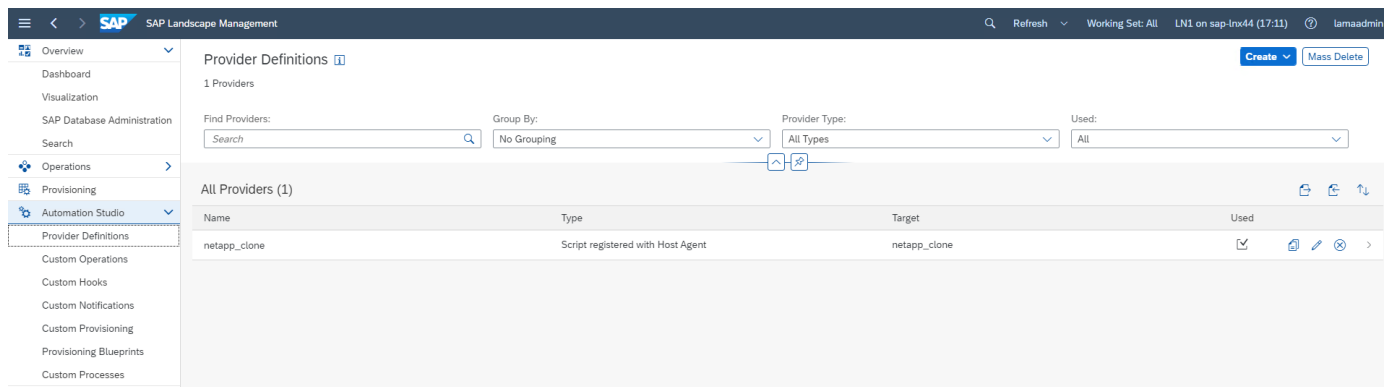
ラボ環境では、次のソフトウェアリリースを使用しました。

- SAP LaMa Enterprise Edition 3.00 SP23_2
- SAP HANA 2.00.052.00.1599235305
- SAP 7.77パッチ27 (S/4 HANA 1909)
- SAPホストエージェント7.22パッチ56
- SAPACEXT 7.22パッチ69
- Linux SLES 15 SP2
- Ansible 2.13.7.
- NetApp ONTAP 9.8P8

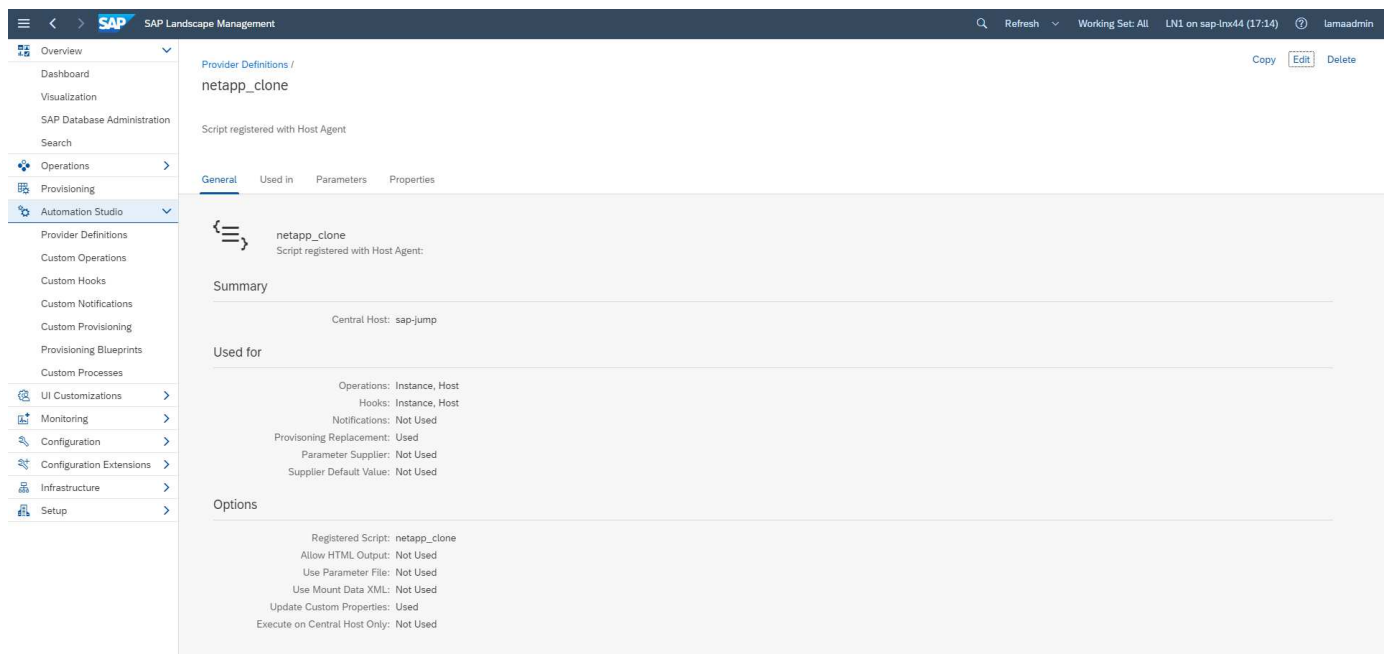
SAP LaMa設定

SAP LaMaプロバイダによる定義

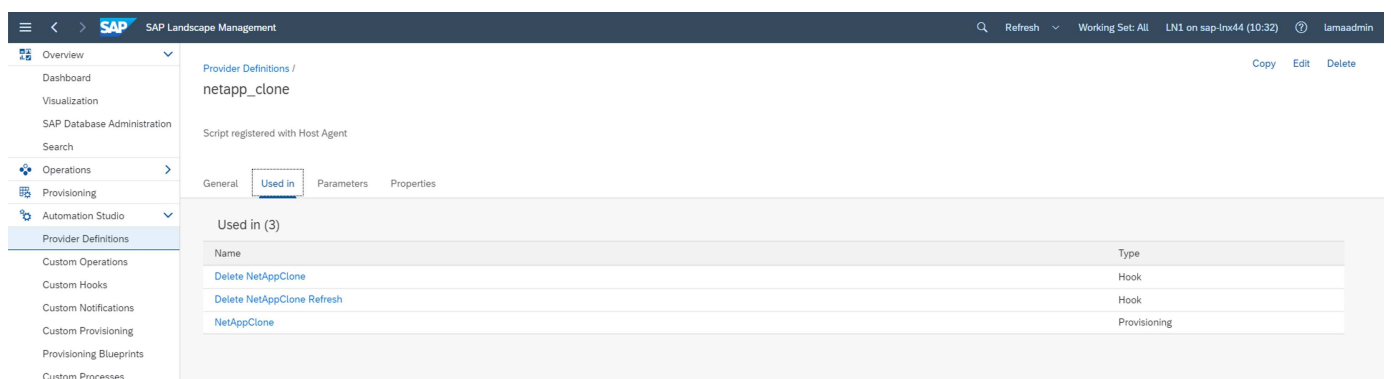
プロバイダの定義は、次のスクリーンショットに示すように、SAP LaMaのAutomation Studio内で実行されます。この例の実装では、前に説明したように、異なるカスタムプロビジョニングステップと操作フックに使用される単一のプロバイダ定義を使用します。



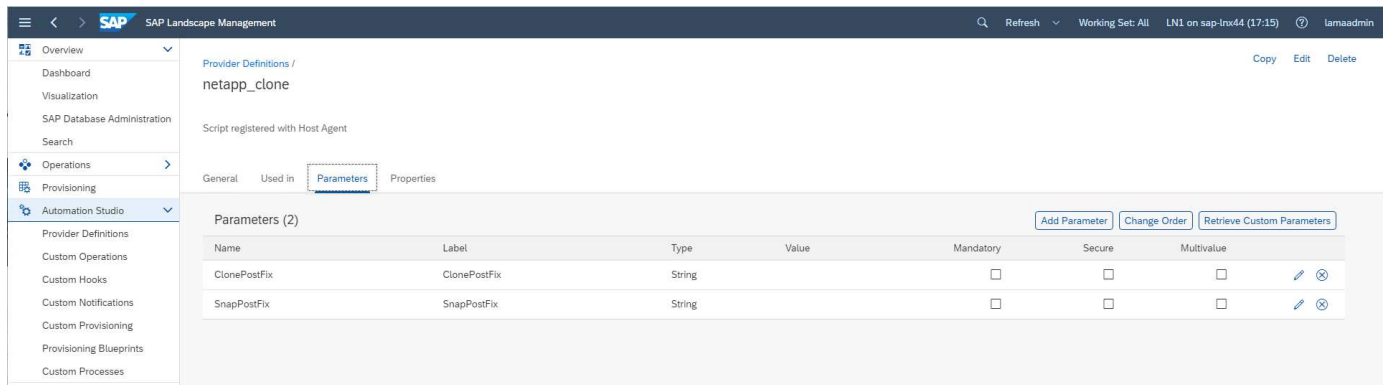
プロバイダ netapp_clone スクリプトとして定義されます netapp_clone.sh SAPホストエージェントに登録されている。SAPホストエージェントは、中央のホストで実行されます `sap-jump` Ansibleコントロールノードとしても機能します。



[で使用（Used in）]タブには、プロバイダーが使用されるカスタムオペレーションが表示されます。カスタムプロビジョニング NetAppClone およびカスタムフック Delete NetAppClone および Delete NetAppClone Refresh *の構成は、次の章で説明します。

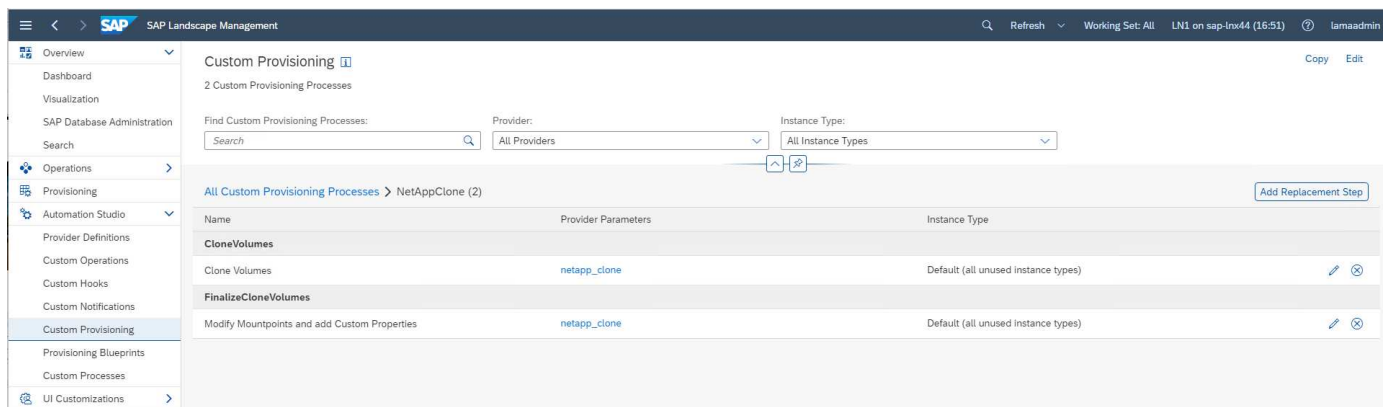


プロビジョニングワークフローの実行中にパラメータ* ClonePostFix および SnapPostFix *が要求され、SnapshotとFlexCloneボリュームの名前に使用されます。



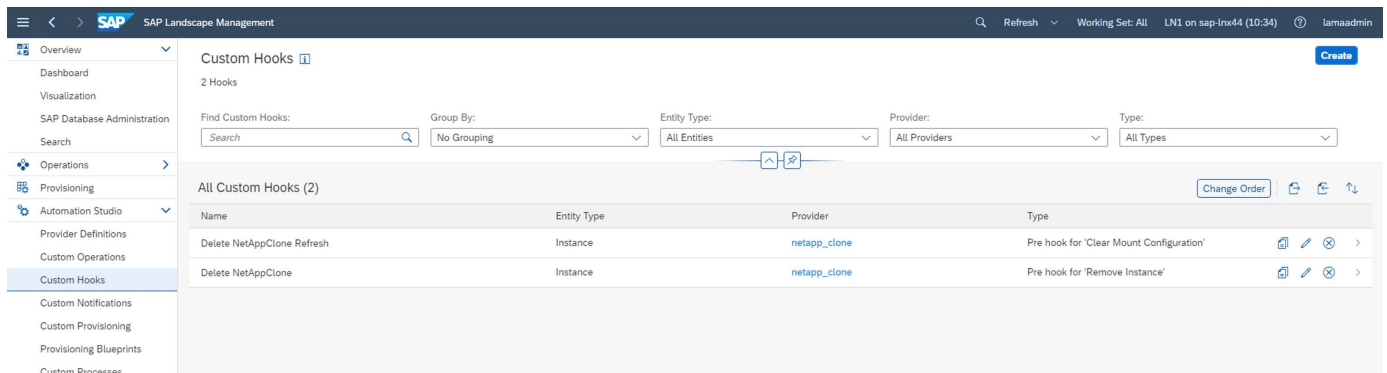
SAP LaMaカスタムプロビジョニング

SAP LaMaのカスタムプロビジョニング設定で、前述したお客様のプロバイダを使用して、プロビジョニングワークフローの手順* Clone Volumes と PostCloneVolumes *を置き換えます。



SAP LaMaカスタムフック

システムの削除ワークフロー時にシステムを削除した場合は、フック* Delete NetAppClone を使用してプロバイダ定義が呼び出されます **netapp_clone**。インスタンスは実行中も保持されるため、システムの更新ワークフロー中は Delete NetApp Clone Refresh *フックが使用されます。



SAP LaMaがマウントポイント設定の情報をプロバイダに提供できるように、カスタムフックに「データマウントXML *を使用」を設定することが重要です。

Custom Hooks /
Delete NetAppClone

Instance

General Parameters Constraints

Summary

Entity Type: Instance
Dynamic Caption:
Hook Type: Pre Hook
Hook for Operation: Remove Instance

Additional Information

Use Mount Data XML: ☒ Yes
Parallel Execution: No
Background Step: No
Process Error Hook: No
Is System Wide Hook: No
Retrieve Secure Parameters: No

カスタムプロビジョニングワークフローを使用してシステムを作成したときにのみカスタムフックが使用および実行されるようにするために、次の制約がそのフックに追加されます。

Custom Hooks /
Delete NetAppClone

Instance

General Parameters Constraints

Constraints (1)

Name	Operator	Value
Custom clone process name (Static)	=	NetAppClone

Add Constraint

カスタムフックの使用方法の詳細については、を参照してください ["SAP LaMaのドキュメント"](#)。

SAPソースシステムでカスタムのプロビジョニングワークフローを有効にします

ソースシステムのカスタムプロビジョニングワークフローを有効にするには、その設定を該当するものにする必要があります。[カスタムプロビジョニングプロセスを使用する*]チェックボックスをオンにして、対応するカスタムプロビジョニング定義を選択する必要があります。

SAP Landscape Management

Automation Studio Configuration Infrastructure

Pools Systems Hosts Characteristics

Overview of Systems and Instances

Discover Remove Instance and System Reassign Instances Mass Configuration Filtering Export Import

Name	Managed	AC-Enabled	Operational	Pool	Network	Description
HN9: NetWeaver ABAP 7.77, cshn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC		
System database: MASTER (configured) : H09, SAP HANA 02, dbh09	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
Central services: 01, cshn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
AS instance: 00, pshn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
HNA: NetWeaver ABAP 7.77, cshn9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	MUCCBC		

Systems: 2 Selected: HN9: NetWeaver ABAP 7.77, cshn9

System Details Log

Edit Show In

General

System Name: HN9: NetWeaver ABAP 7.77, cshn9

SID: HN9

Instance ID: SystemID HN9, SystemHost cshn9

Solution Manager settings

Assign Solution Manager System:

Focused Run Settings

Assign Focused Run System:

Disable Workmode Management:

System and AS Provisioning

This system was provided by:

This system can be used for:

Installation

☒ Cloning ☐ Application Server (Un-)Installation

☐ Copying ☐ Diagnostic Agent (Un-)Installation

☐ Renaming ☐ nZDM Java

☐ Standalone PCA ☐ Replication Configuration

Use Custom Provisioning Process:

☒ NetAppClone

Use as TMS Control System:

Is BW Source System:

Use Replication for Single Tenant Database Refresh:

Intersystem Dependencies

From Instance To Instance

Outgoing (0)

Incoming (0)

Entity Relations

Custom Relation Type Target Entity Type Target Entity

Table is empty

E-Mail Notification

Enable Email Notification:

Custom Notification

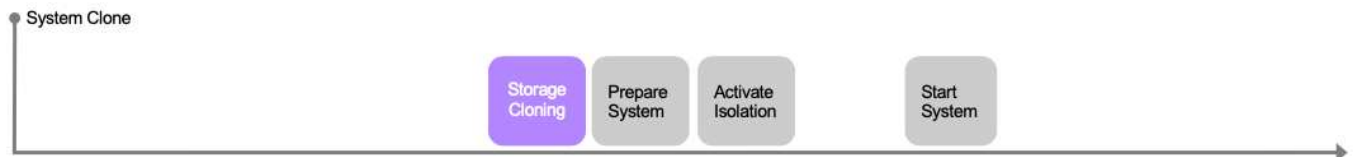
Enable Custom Notification:

ACM Settings

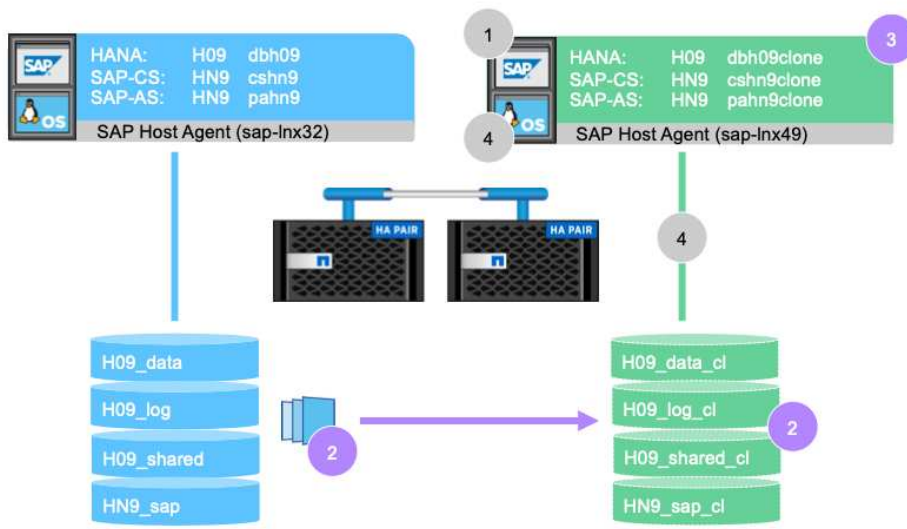
ACM-Managed:

SAP LaMaプロビジョニングワークフロー-クローンシステム

次の図は、システムのクローニングワークフローで実行する主な手順を示しています。



このセクションでは、HANAデータベースH09を使用するソースのSAPシステムHN9に基づく、SAP LaMaシステムの完全なクローニングワークフローを紹介します。次の図は、ワークフロー中に実行される手順の概要を示しています。



SAP LaMa (sap-lnx44)

Ansible Control Node
 netapp_clone.sh

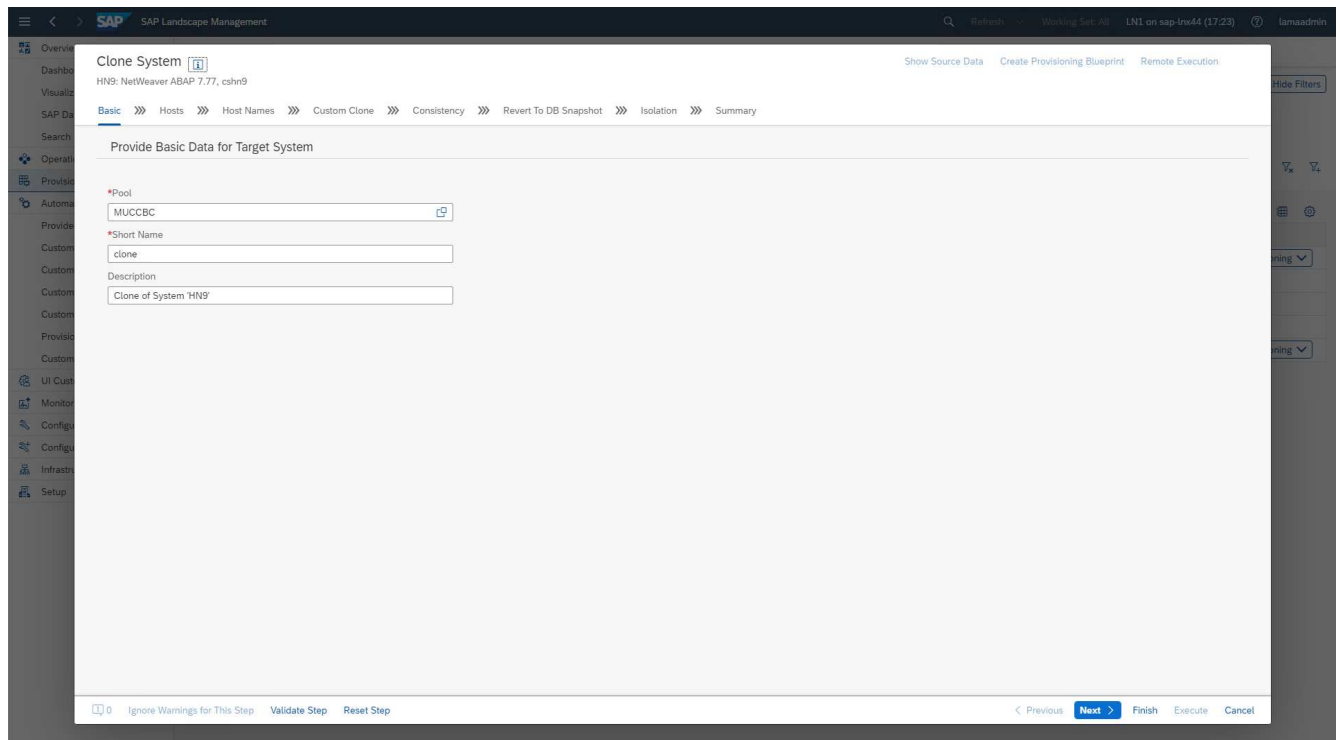
SAP Host Agent (sap-jump)

- 1 Create Cloned System Configuration
- 2 Create Storage Snapshot + Clone (netapp_lama_CloneVolumes.yml)
- 3 Create Mount Point Configuration + Set Custom Properties
- 4 Prepare + Start System

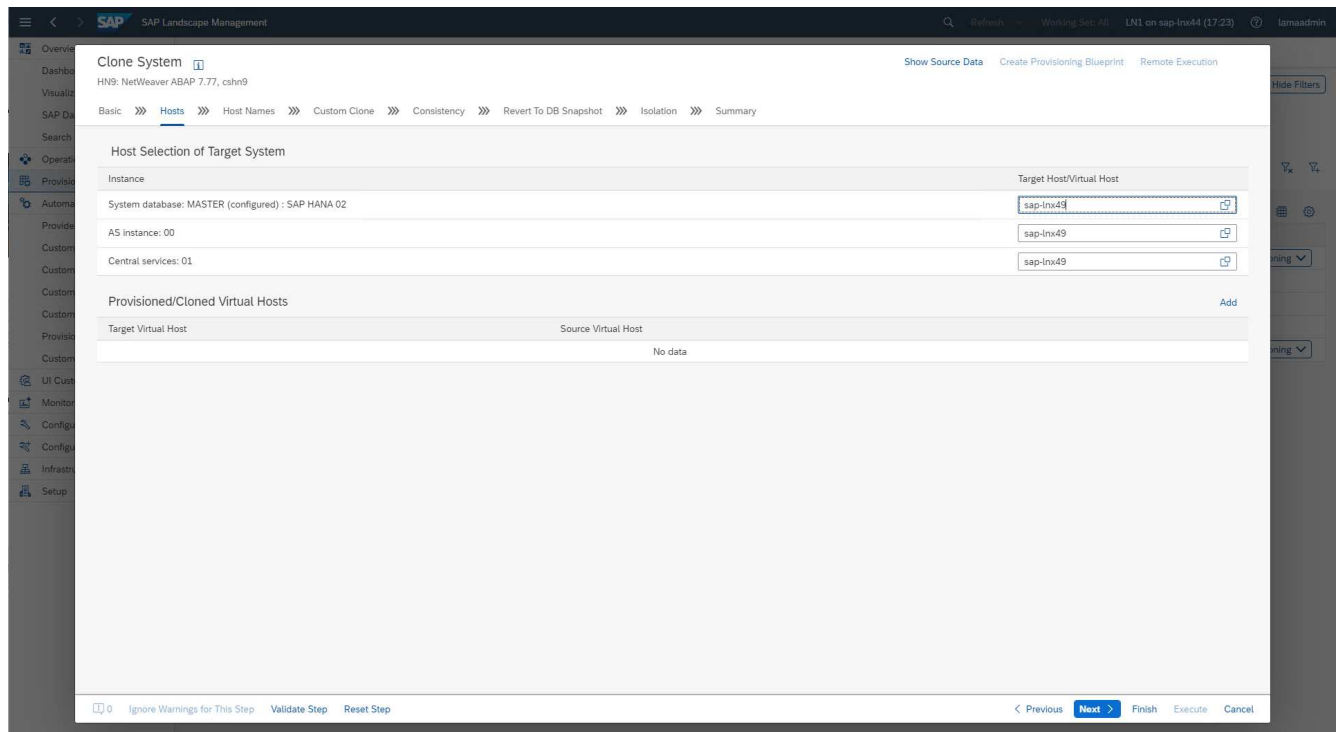
1. クローニングワークフローを開始するには、メニューツリーで* Provisioning を開き、ソースシステム（この例ではHN9）を選択します。次に、Clone System *ウィザードを起動します。

Name	Pool	Description	Assigned Host	Virtualized	Provisioning
HN9: NetWeaver ABAP 7.77, cshn9	MUCCBC				Provisioning
H09 System database (ABAP): MASTER : SAP HANA 02, dbh09	MUCCBC		sap-lnx32		
HN9 Central services (ABAP): 01, cshn9	MUCCBC		sap-lnx32		
HN9 AS instance (ABAP): 00, pahn9	MUCCBC		sap-lnx32		
HNA: NetWeaver ABAP 7.77, cshna	MUCCBC				Provisioning

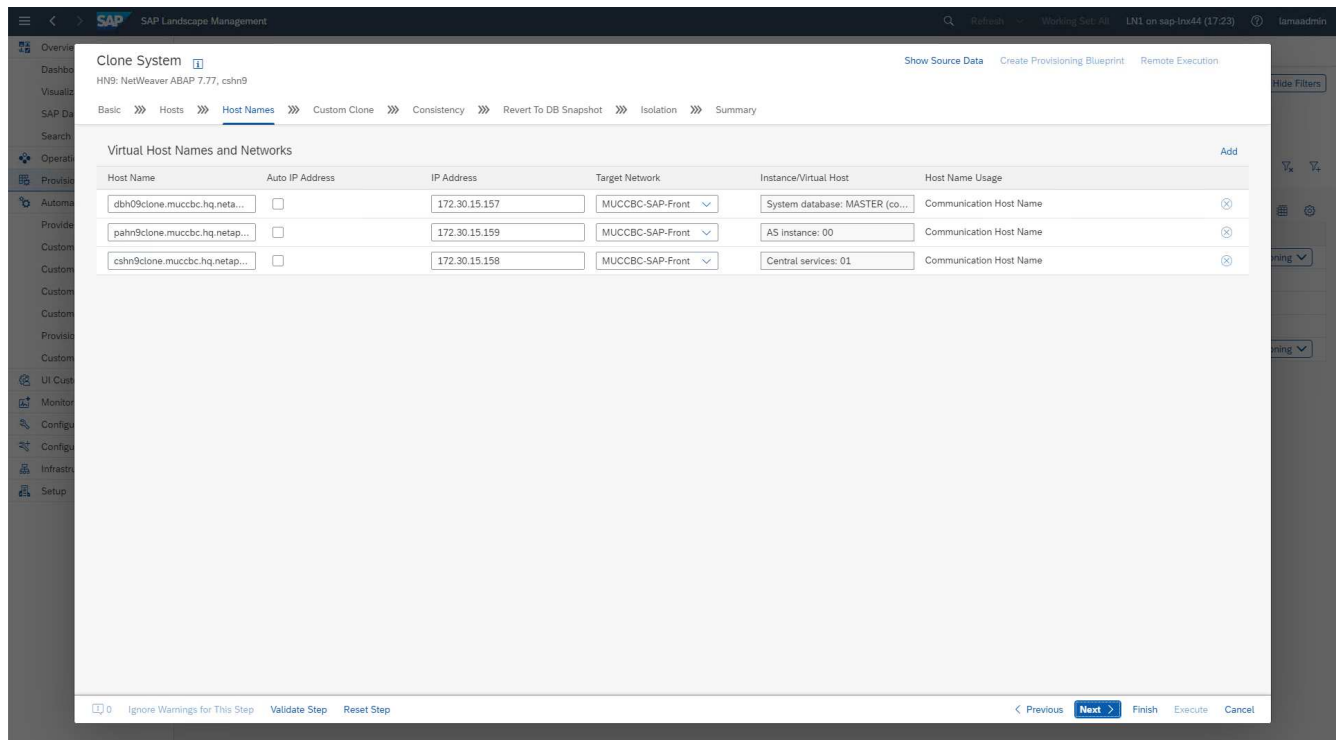
2. 必要な値を入力します。ウィザードの画面1で、クローニングしたシステムのプール名を入力するように求められます。このステップでは、クローニングされたシステムを起動するインスタンス（仮想または物理）を指定します。デフォルトでは、システムはターゲットシステムと同じプールにクローニングされます。



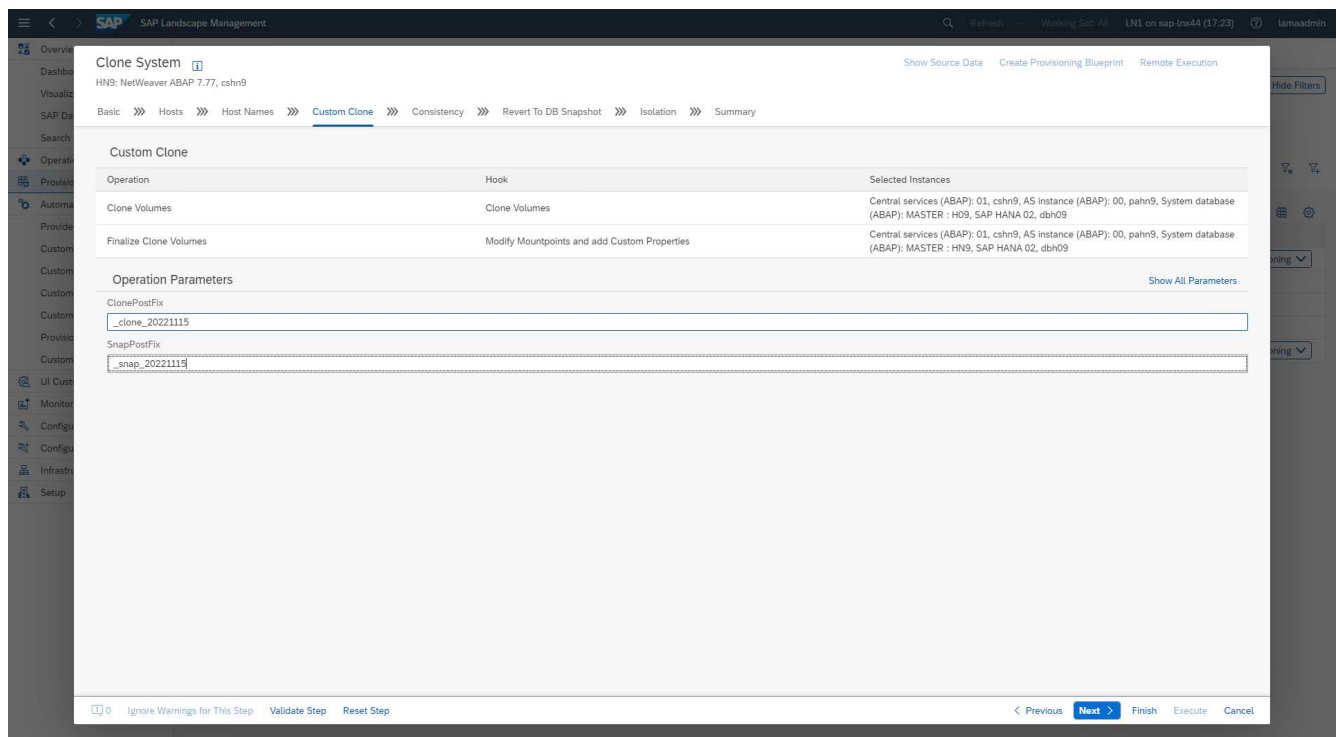
3. ウィザードの画面2で、新しいSAPインスタンスを起動するターゲットホストを指定するよう求められます。このインスタンスのターゲットホストは、前の画面で指定したホストプールから選択できます。各インスタンスまたはサービスは、別々のホストで開始できます。この例では、3つのサービスがすべて同じホスト上で実行されています。



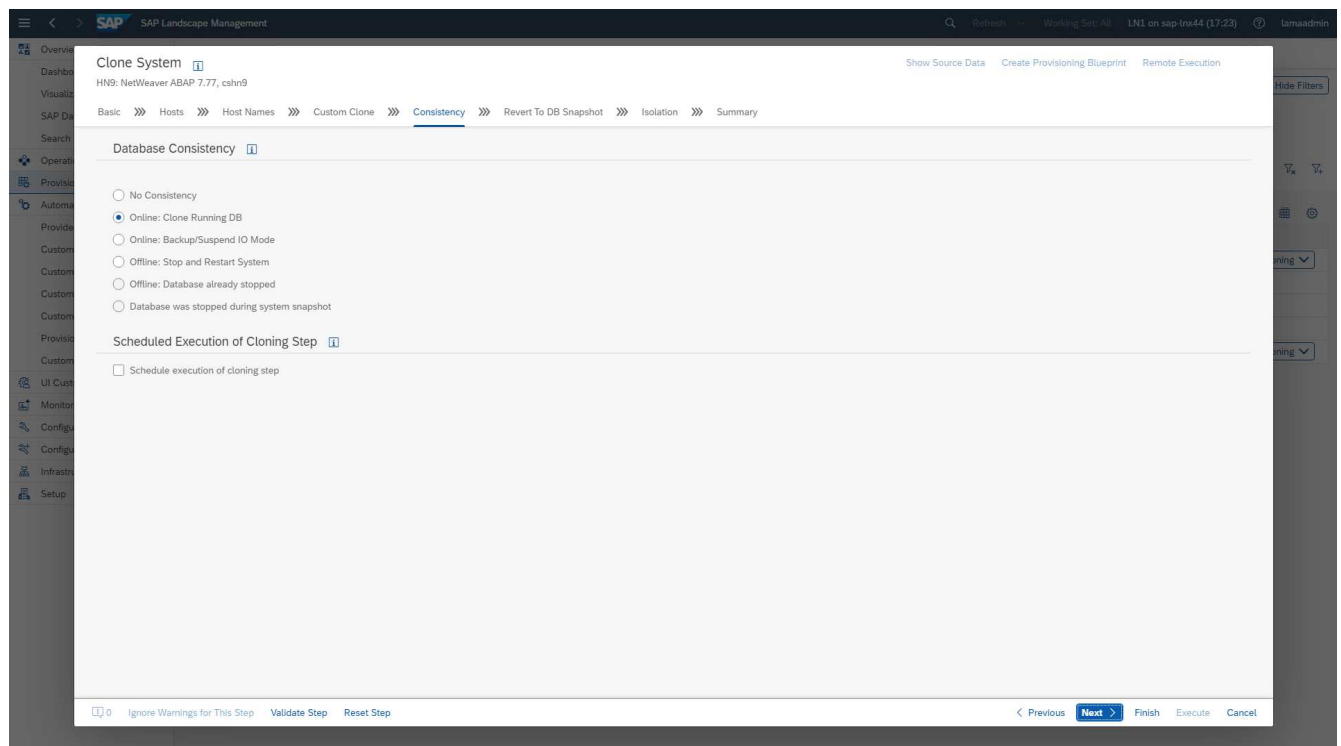
4. 画面3で要求された情報を入力します。この情報は、仮想ホスト名とネットワークを要求します。通常、ホスト名はDNSで管理されるため、IPアドレスもそれに応じて事前に入力されます。



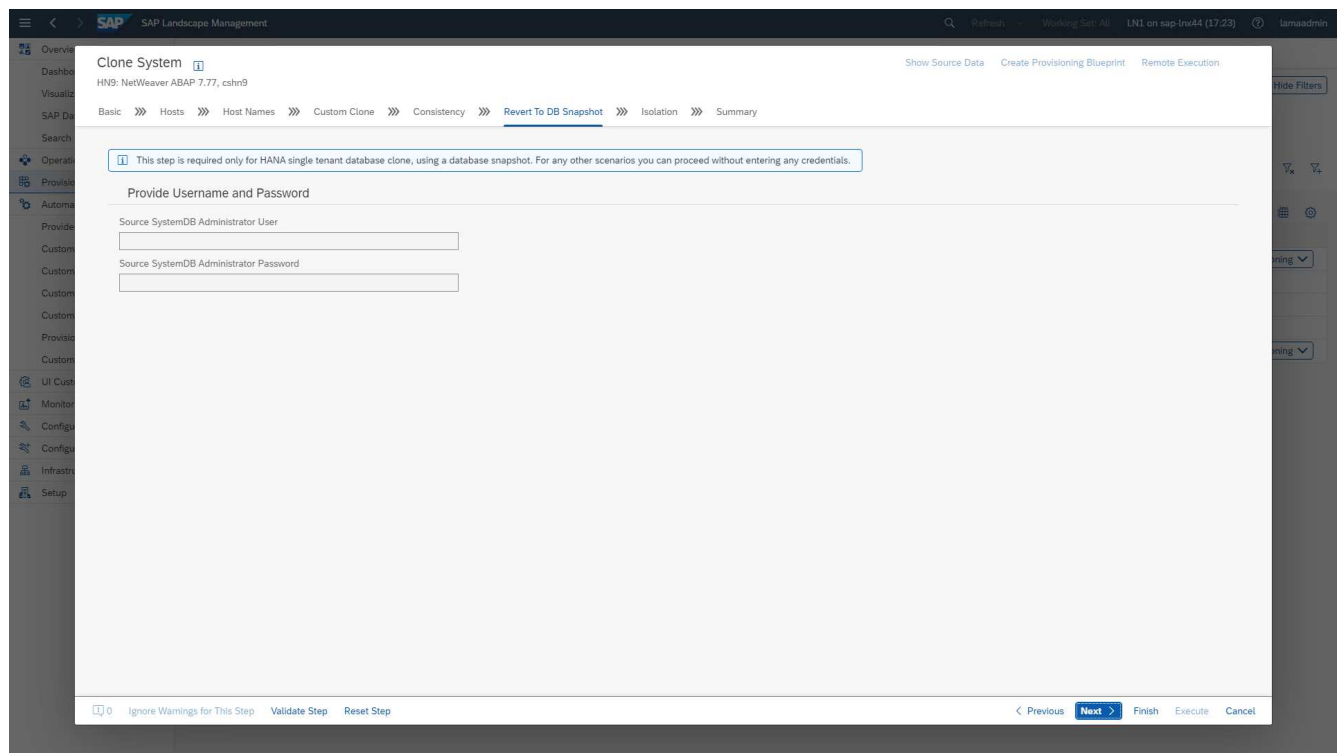
5. 画面4に、カスタムのクローン処理が表示されます。クローン名と* SnapPostfix *名を指定した場合、この名前は、FlexCloneボリュームとSnapshot名に対するストレージのクローン処理でそれぞれ使用されます。これらのフィールドを空のままにしておくと、プロバイダスクリプトの変数セクションで設定されたデフォルト値になります netapp_clone.sh を使用します。



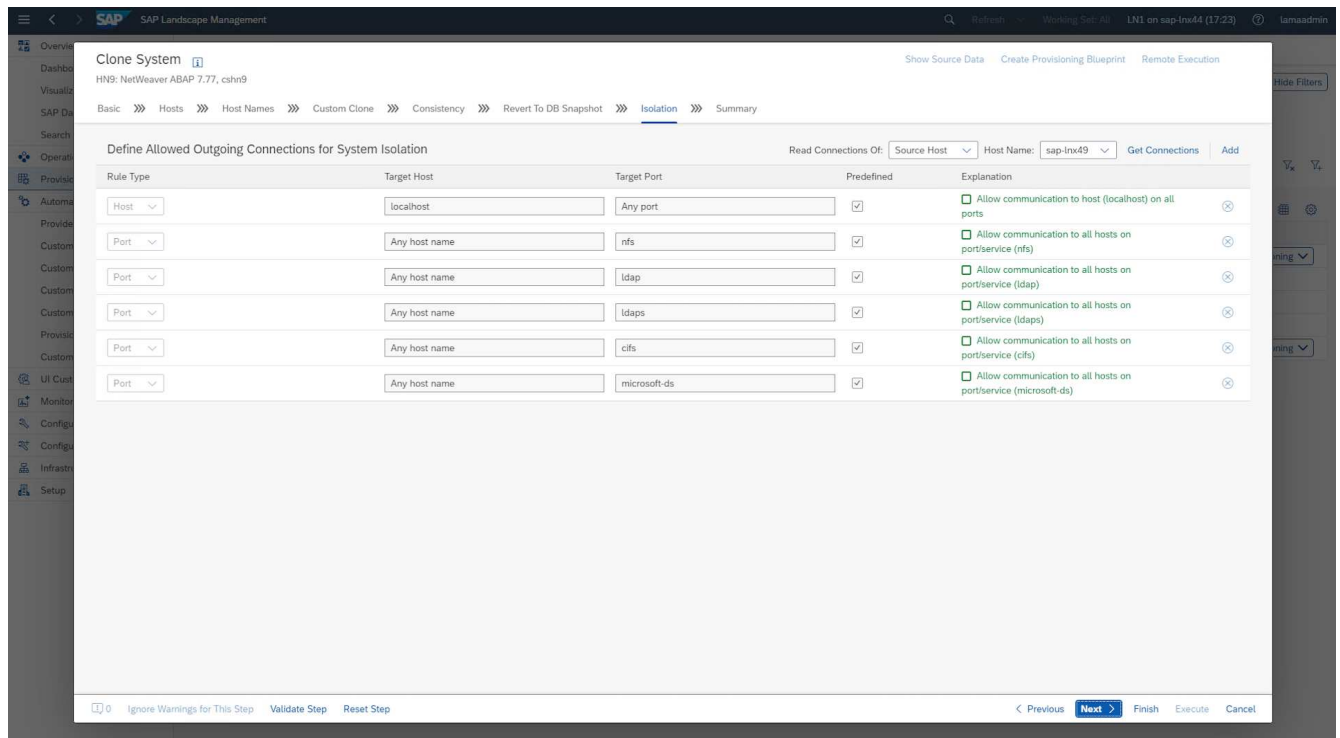
6. 画面5では、データベース整合性オプションが選択されています。この例では、「* Online : Clone running DB *」を選択しました。



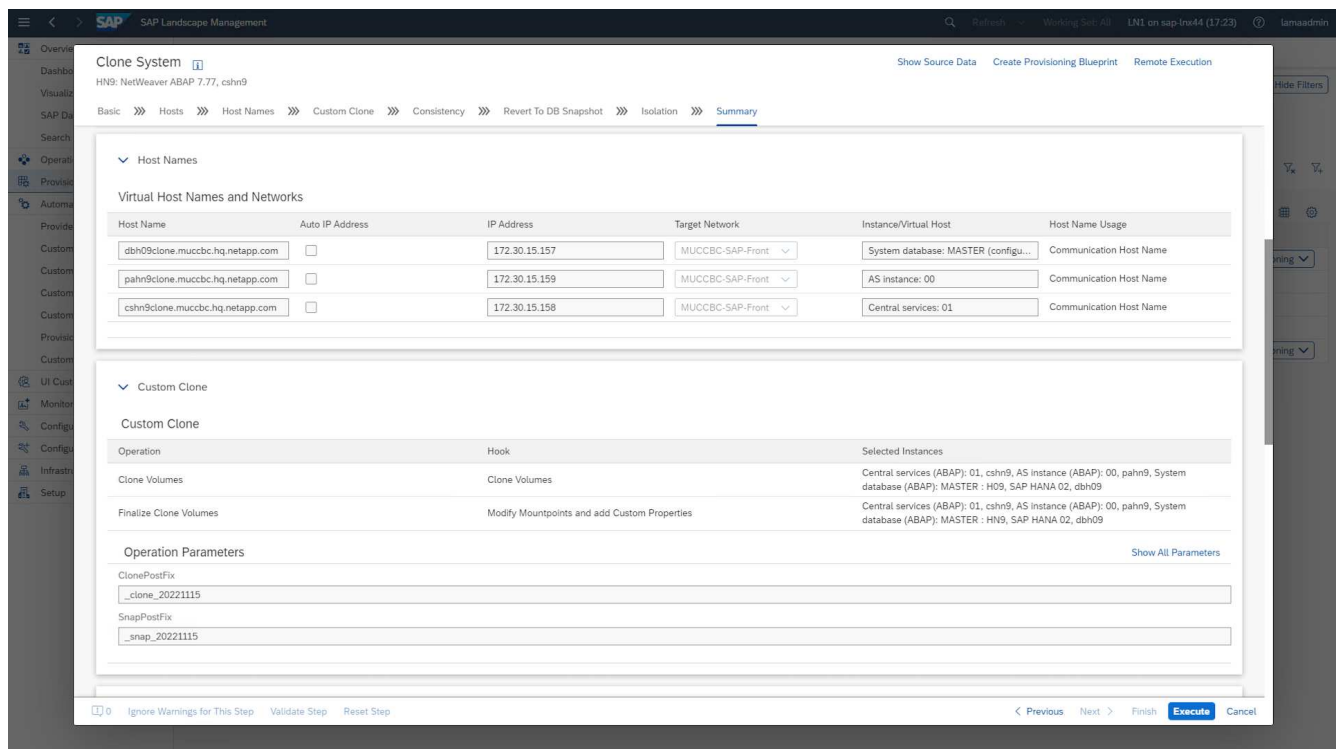
7. 画面6では、テナントのクローンを実行する場合にのみ入力する必要があります。



8. 画面7では、システムの隔離を構成できます。

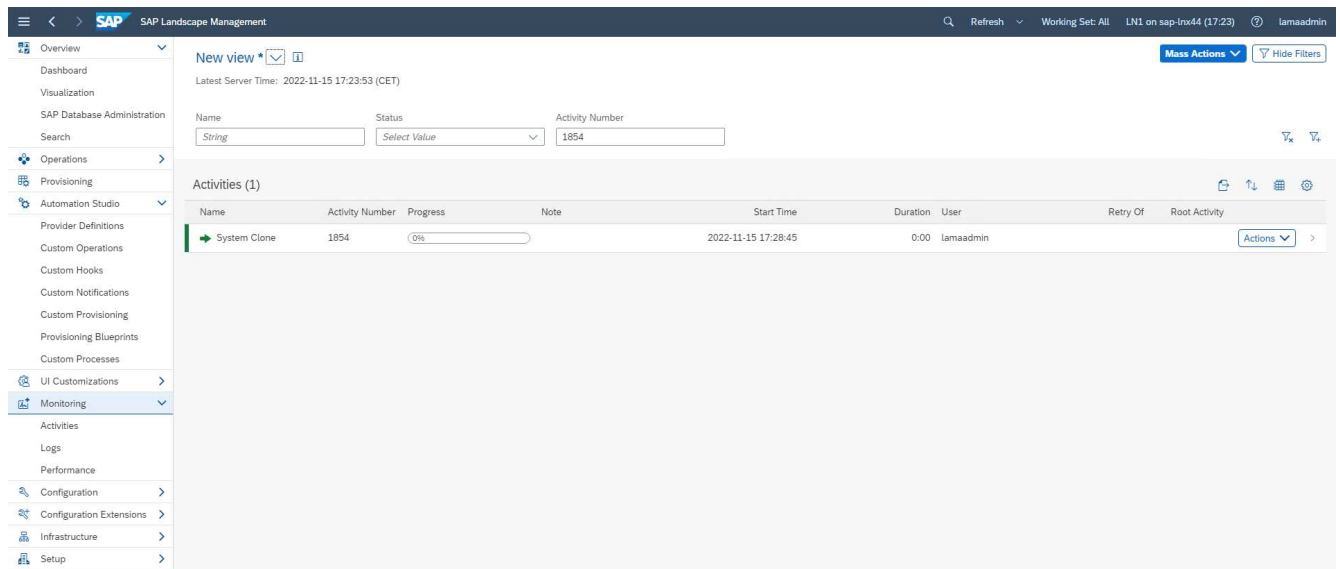


9. 画面8では、ワークフローを開始する前の最終確認のためのすべての設定が概要ページに表示されます。
[Execute]をクリックして'ワークフローを開始します'

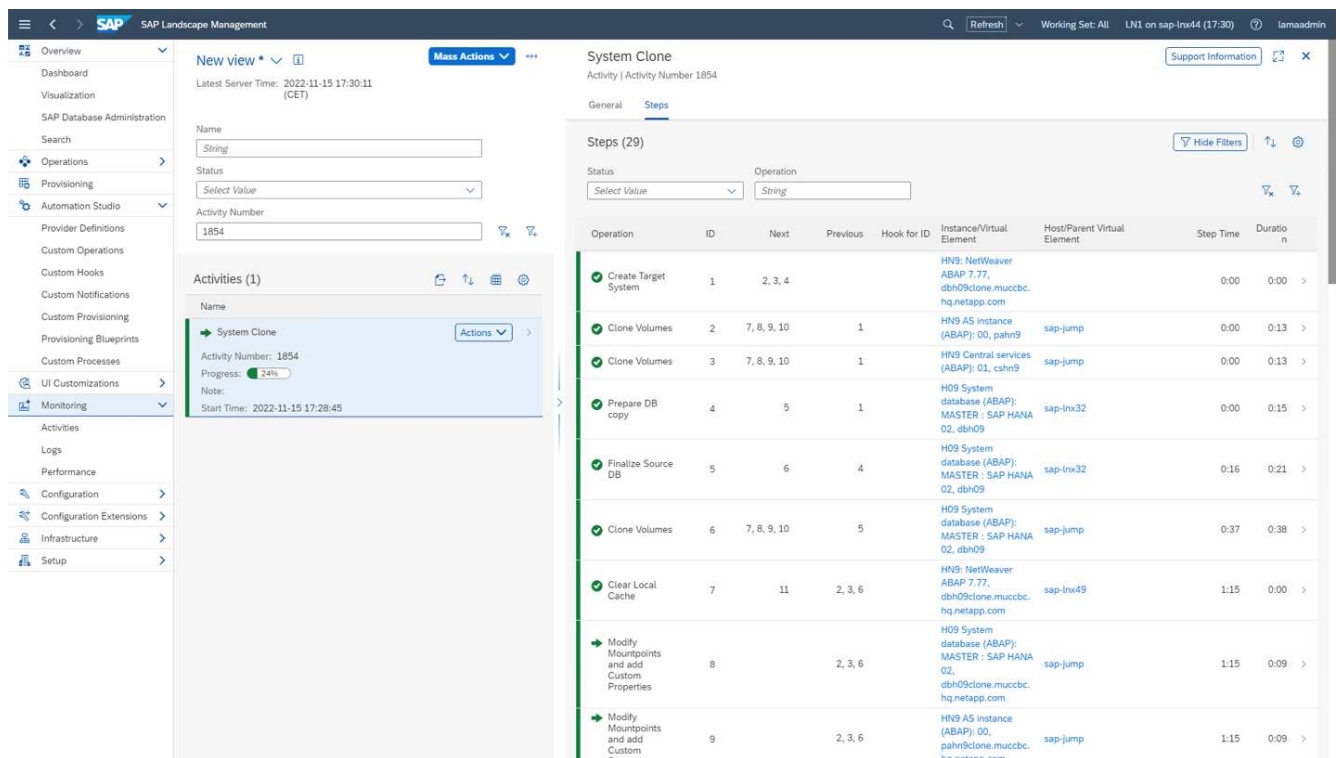


SAP LaMaが、設定に記載されたすべてのアクションを実行します。具体的には、ストレージボリュームのクローンとエクスポートの作成、ターゲットホストへのマウント、分離のためのファイアウォールルールの追加、HANAデータベースとSAPサービスの開始です。

10. クローンワークフローの進行状況は、* Monitoring *メニューで監視できます。



詳細ログには、* Clone Volume および ModifyマウントポイントとAdd Custom Properties *の処理がAnsibleノード、で実行されます sap-jump ホスト：以下の手順は、各サービス、HANAデータベース、SAP中央サービス、およびSAPサービスに対して実行します。



- Clone Volumes *タスクを選択すると、そのステップの詳細ログが表示され、Ansible Playbookの実行状況が表示されます。Ansibleプレイブックも確認できます netapp_lama_CloneVolumes.yml は、HANAデータベースのボリューム、データ、ログ、および共有ごとに実行されます。

The screenshot shows the SAP Landscape Management interface. The left sidebar contains navigation options like Overview, Dashboard, Visualization, SAP Database Administration, Search, Operations, Provisioning, Automation Studio, Provider Definitions, Custom Operations, Custom Hooks, Custom Notifications, Custom Provisioning, Provisioning Blueprints, Custom Processes, UI Customizations, Monitoring, Activities, Logs, and Performance. The main area displays the 'System Clone' activity. The 'Clone Volumes' step is selected, and the 'Messages' pane shows a list of messages. A red box highlights the messages related to the 'Clone Volumes' step, including 'Running ansible playbook netapp_lama_CloneVolumes.yml on Volume H09_shared' and 'Running ansible playbook netapp_lama_CloneVolumes.yml on Volume H09_log'.

12. マウントポイントの変更とカスタムプロパティの追加の手順*マウントポイントの詳細ビューには、マウントポイントと、実行スクリプトによって渡されたカスタムプロパティに関する情報が表示されます。

The screenshot shows the SAP Landscape Management interface. The left sidebar contains navigation options like Overview, Dashboard, Visualization, SAP Database Administration, Search, Operations, Provisioning, Automation Studio, Provider Definitions, Custom Operations, Custom Hooks, Custom Notifications, Custom Provisioning, Provisioning Blueprints, Custom Processes, UI Customizations, Monitoring, Activities, Logs, and Performance. The main area displays the 'System Clone' activity. The 'Modify Mountpoints and add Custom Properties' step is selected, and the 'Messages' pane shows a list of messages. A red box highlights the messages related to the 'Modify Mountpoints and add Custom Properties' step, including 'Got new property SnapPostFix_snap_20221115' and 'Got new property ClonePostFix_clone_20221115'.

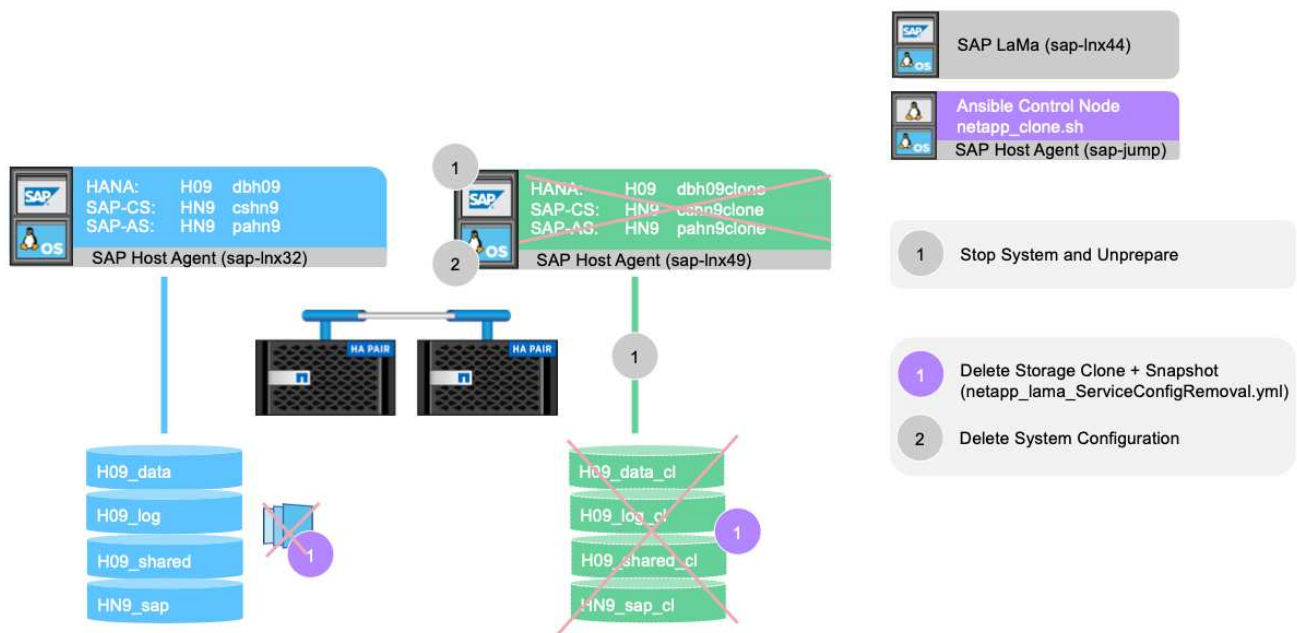
ワークフローが完了すると、クローニングしたSAPシステムの準備が完了し、使用できる状態になります。

SAP LaMaによるワークフローのプロビジョニング解除-システムの破壊

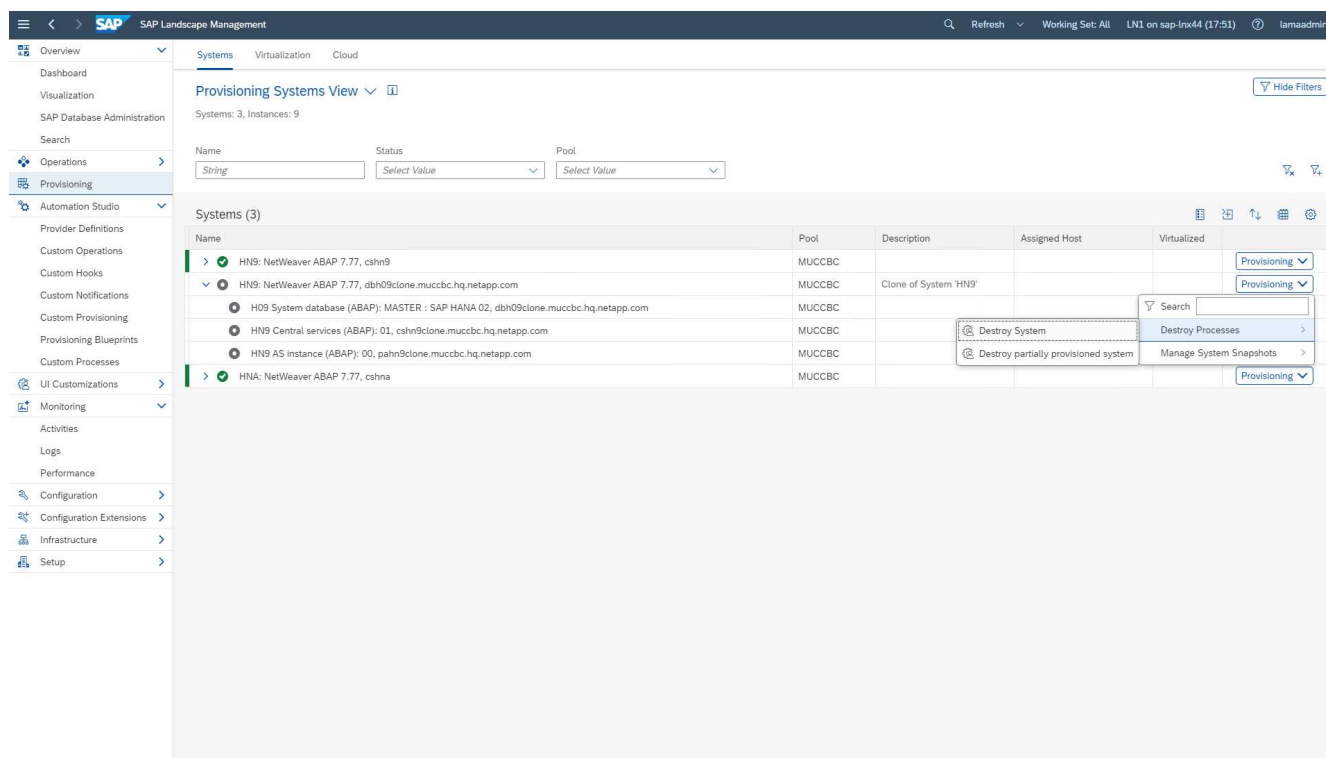
次の図は、system destroyワークフローで実行する主な手順を示しています。



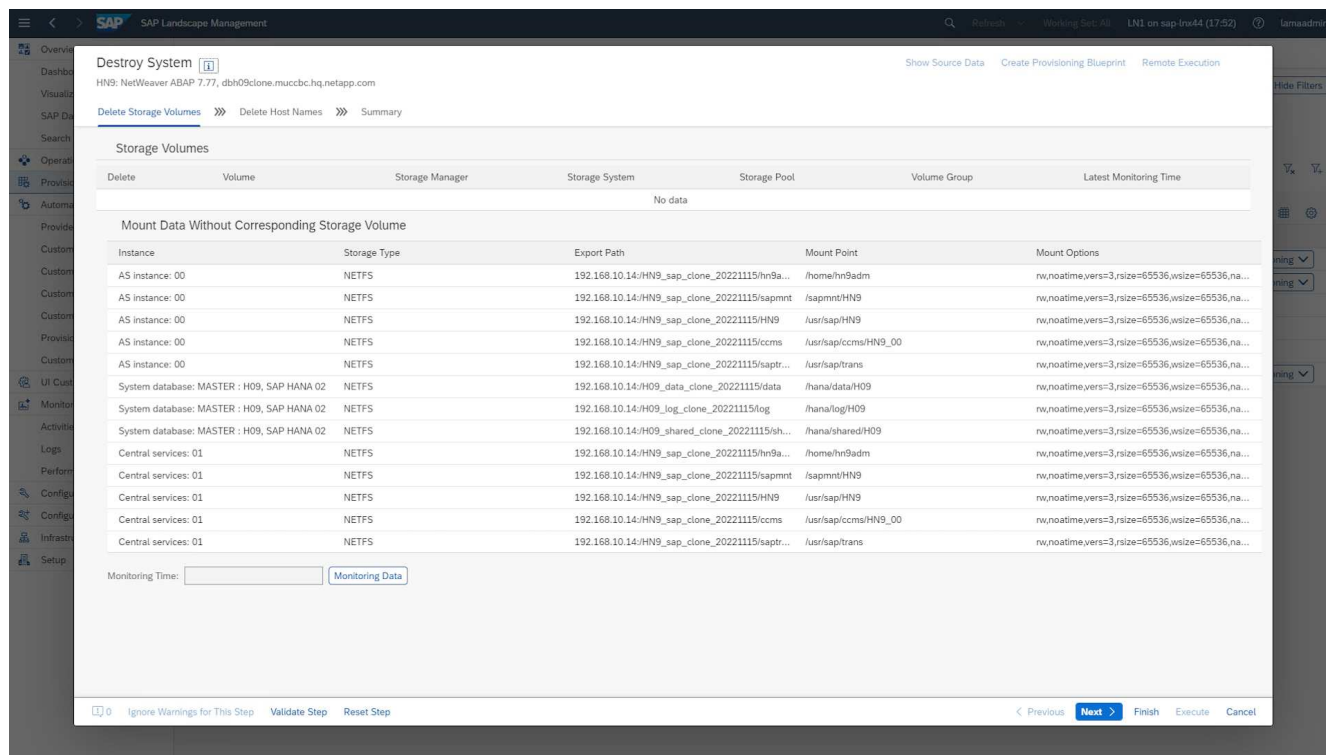
1. クローンシステムの運用を停止するには、運用を停止し、準備を事前に完了しておく必要があります。その後、システムの破壊ワークフローを開始できます。



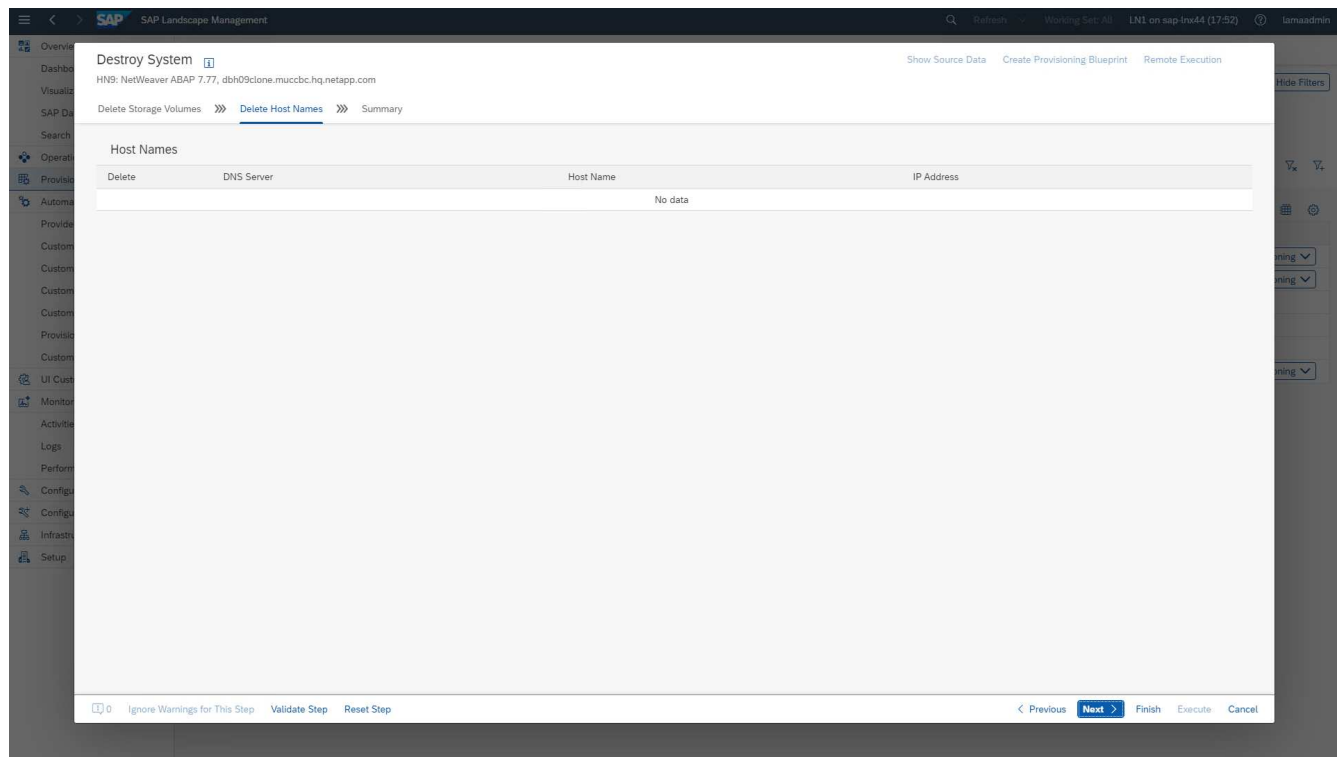
2. この例では、前に作成したシステムに対してsystem destroyワークフローを実行します。システムビュー*画面でシステムを選択し、*破壊プロセス*でシステム破壊ワークフローを開始します。



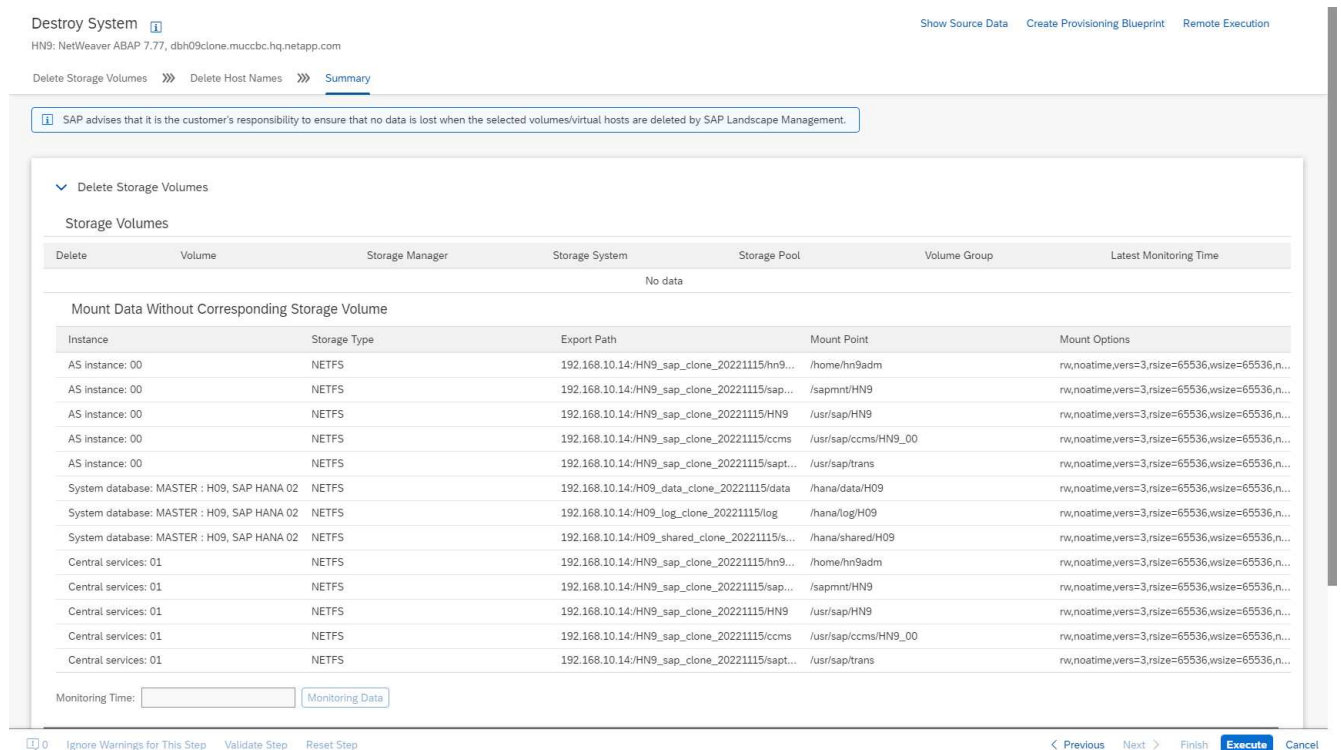
3. プロビジョニングフェーズで維持されたすべてのマウントポイントが表示され、システムの削除ワークフロープロセス中に削除されます。



仮想ホスト名はDNSを通じて管理され、自動的に割り当てられているため、削除されません。

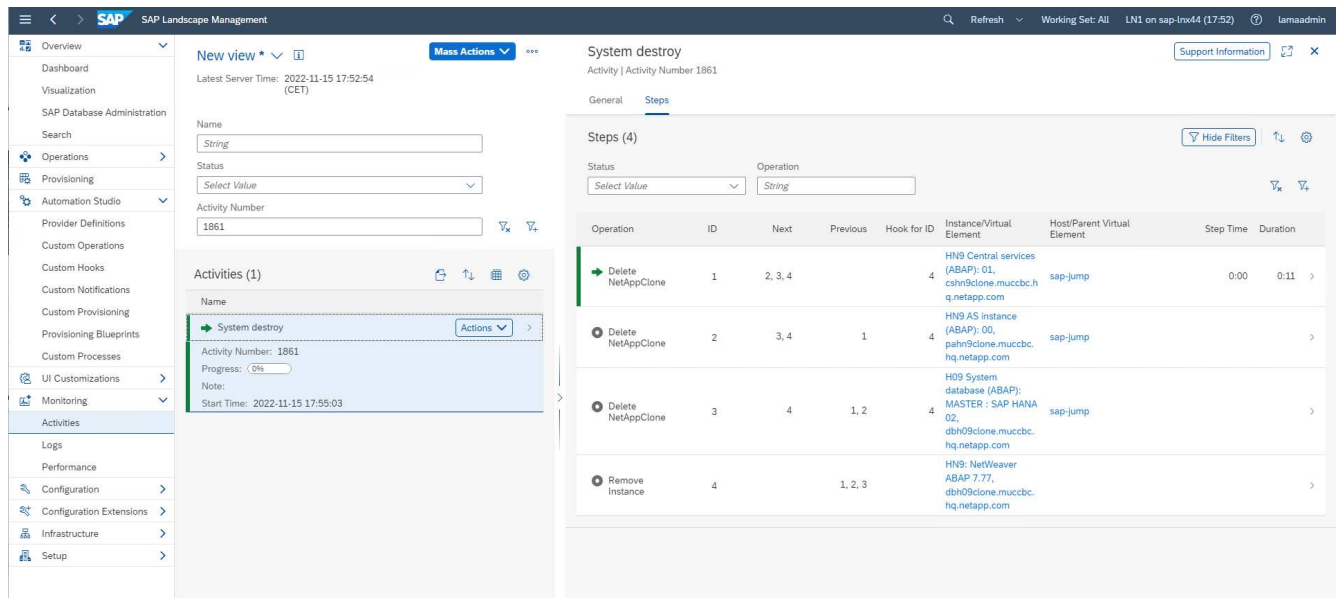


4. 実行ボタンをクリックすると、処理が開始されます。

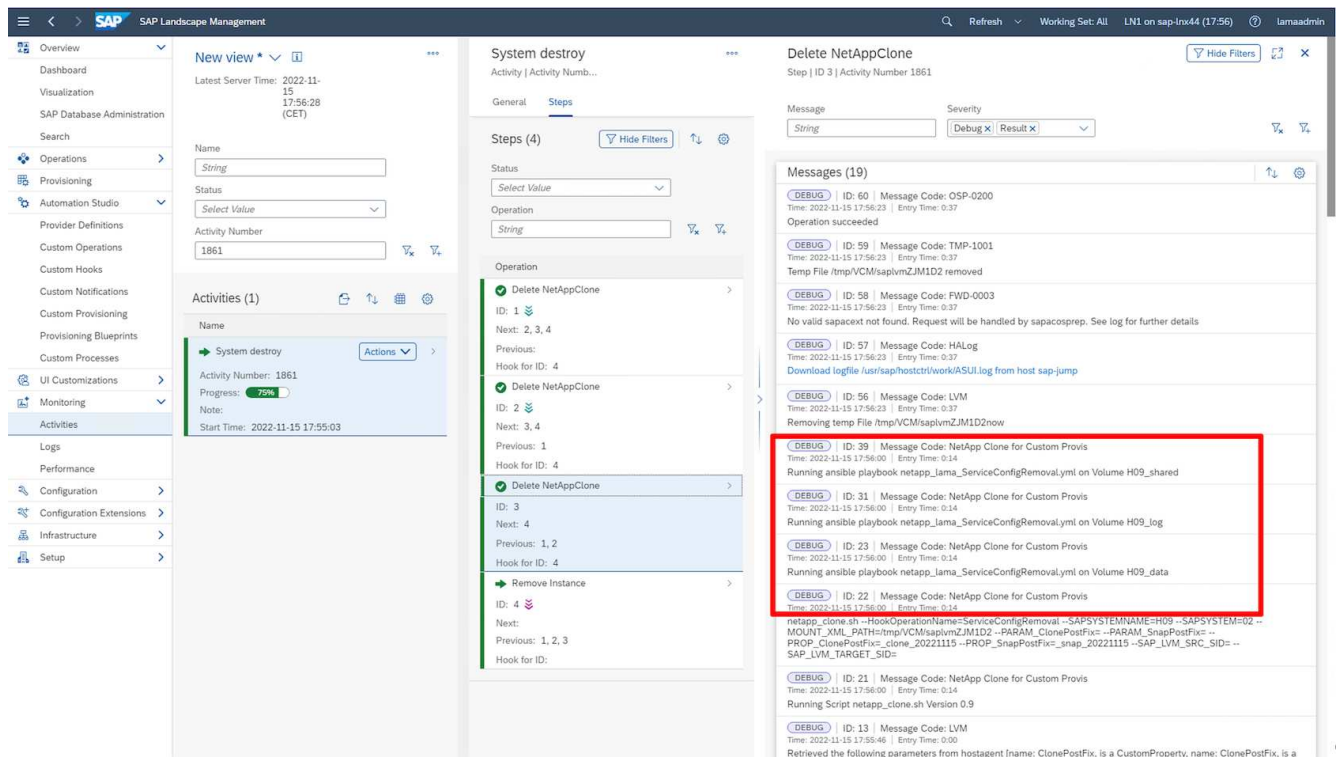


ボリュームクローンを削除し、そのクローンシステムの設定を削除する処理がSAP LaMaから実行されます。

5. クローンワークフローの進行状況は、* Monitoring *メニューで監視できます。

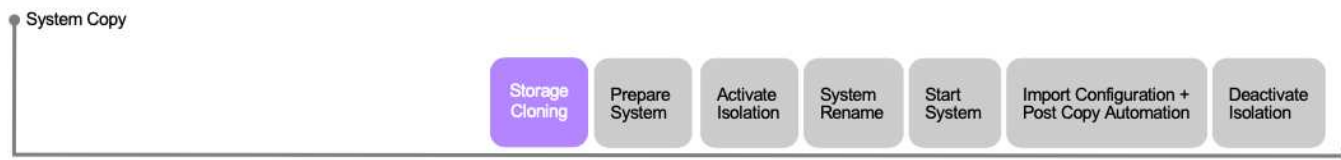


6. 「* Delete NetAppClone *」タスクを選択すると、その手順の詳細ログが表示されます。Ansibleプレイブブックの実行方法を以下に示します。ご覧のように、Ansibleプレイブブックもご覧いただけます
netapp_lama_ServiceConfigRemoval.yml は、HANAデータベースのボリューム、データ、ログ、および共有ごとに実行されます。

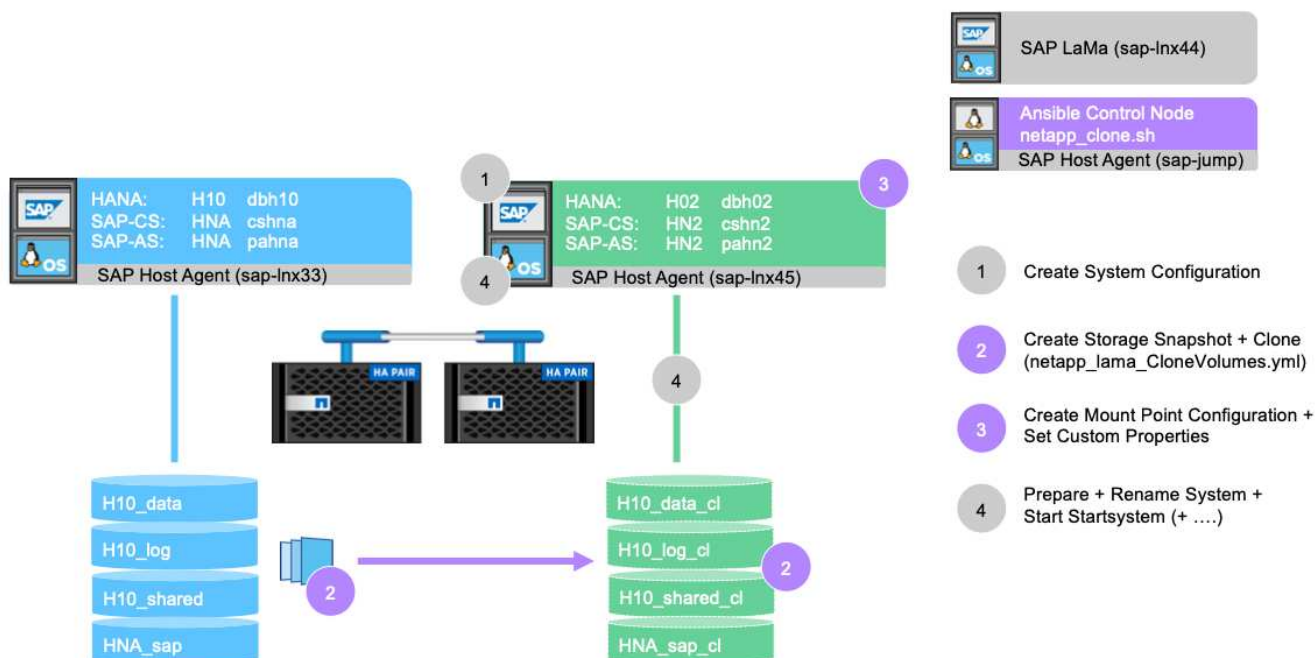


SAP LaMaプロビジョニングワークフロー-コピーシステム

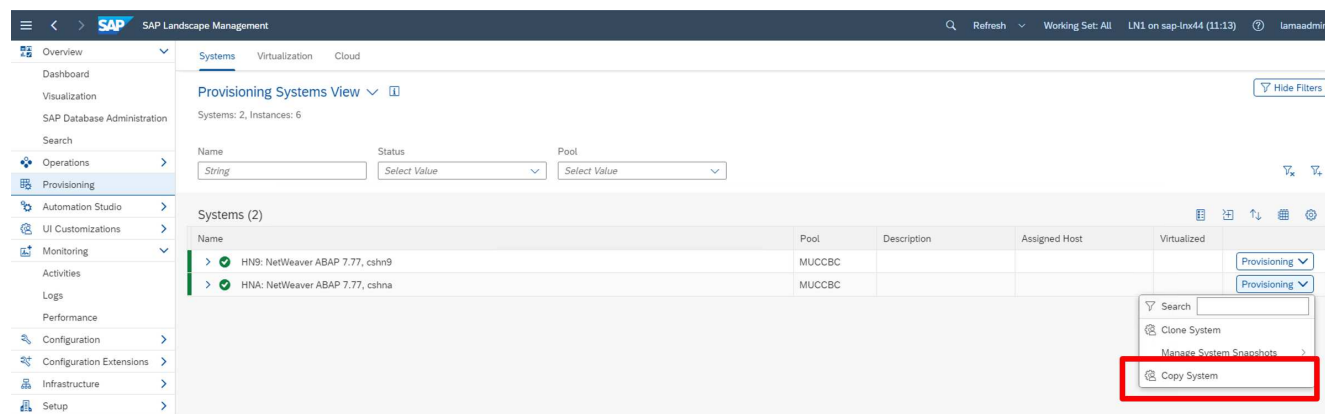
次の図は、システムコピーワークフローで実行する主な手順を示しています。



この章では、システムクローンのワークフローと入力画面の違いについて簡単に説明します。次の図に示すように、ストレージワークフローは変更されていません。



1. システムの準備が整ったら、システムコピーワークフローを開始できます。これは、この設定に関する特定のタスクではなく、詳細な説明も行いません。詳細については、SAP LaMaのドキュメントをご覧ください。



ドロップダウンが強調表示されます。"]

2. コピーワークフローの実行中、最初の画面で指定する必要があるのと同様に、システムの名前が変更されます。

Copy System HN2

HNA: NetWeaver ABAP 7.77, csna

Basic >>> Hosts >>> Host Names >>> Instance Number >>> Custom Clone >>> Consistency >>> Users >>> Rename >>> Isolation >>> ABAP PCA >>> Summary

Provide Basic Data for Target System

*System ID: HN2

☒ Use different Database Name

*HANA SID: H02

*Pool: MUCCBC

Description: Copy of System 'HNA'

Set Master Password for OS and DB Users

*Password: *****

*Confirm Password: *****

Ignore Warnings for This Step Validate Step Reset Step

< Previous **Next** > Finish Execute Cancel

3. ワークフロー中にインスタンス番号を変更することができます。

Copy System HN2

HNA: NetWeaver ABAP 7.77, csna

Basic >>> Hosts >>> Host Names >>> Instance Number >>> Custom Clone >>> Consistency >>> Users >>> Rename >>> Isolation >>> ABAP PCA >>> Summary

SAP Instance Numbers

*System database: MASTER (configured) : SAP HANA 02: 02

*AS instance: 00: 00

*Central services: 01: 01

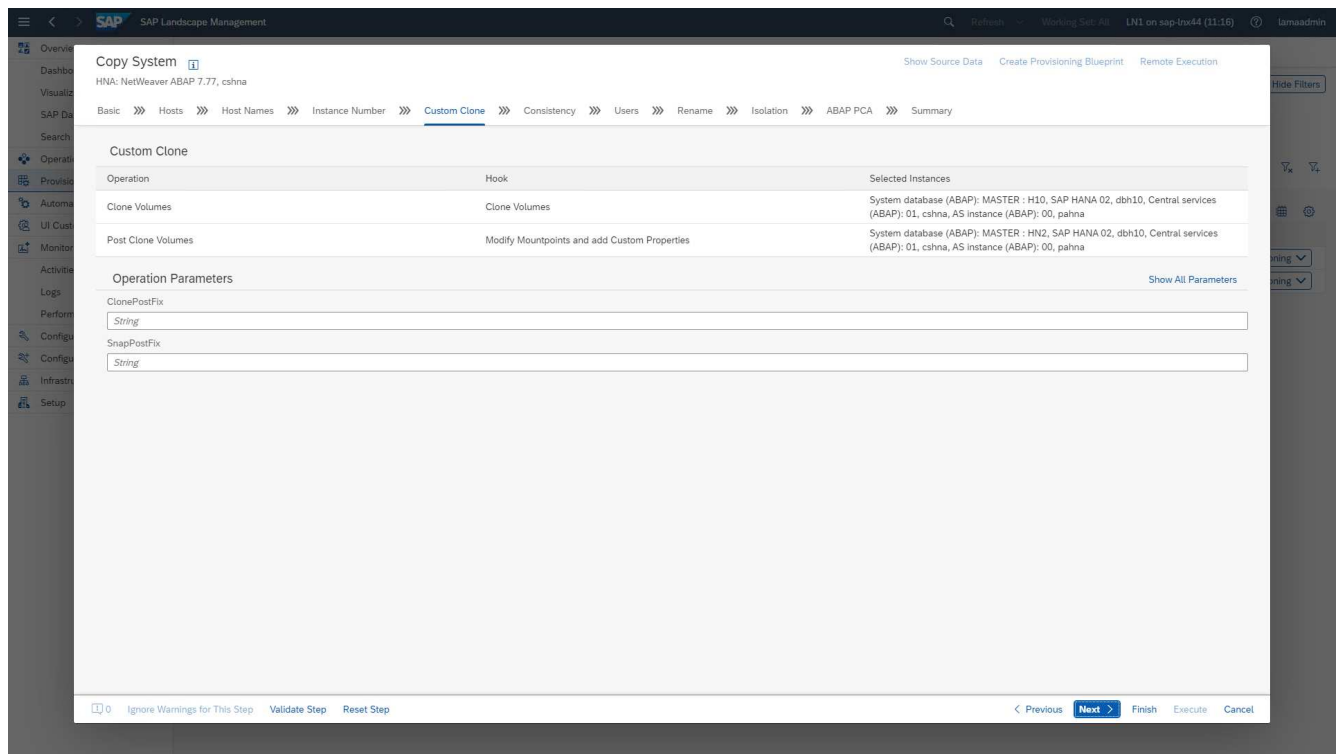
Ignore Warnings for This Step Validate Step Reset Step

< Previous **Next** > Finish Execute Cancel

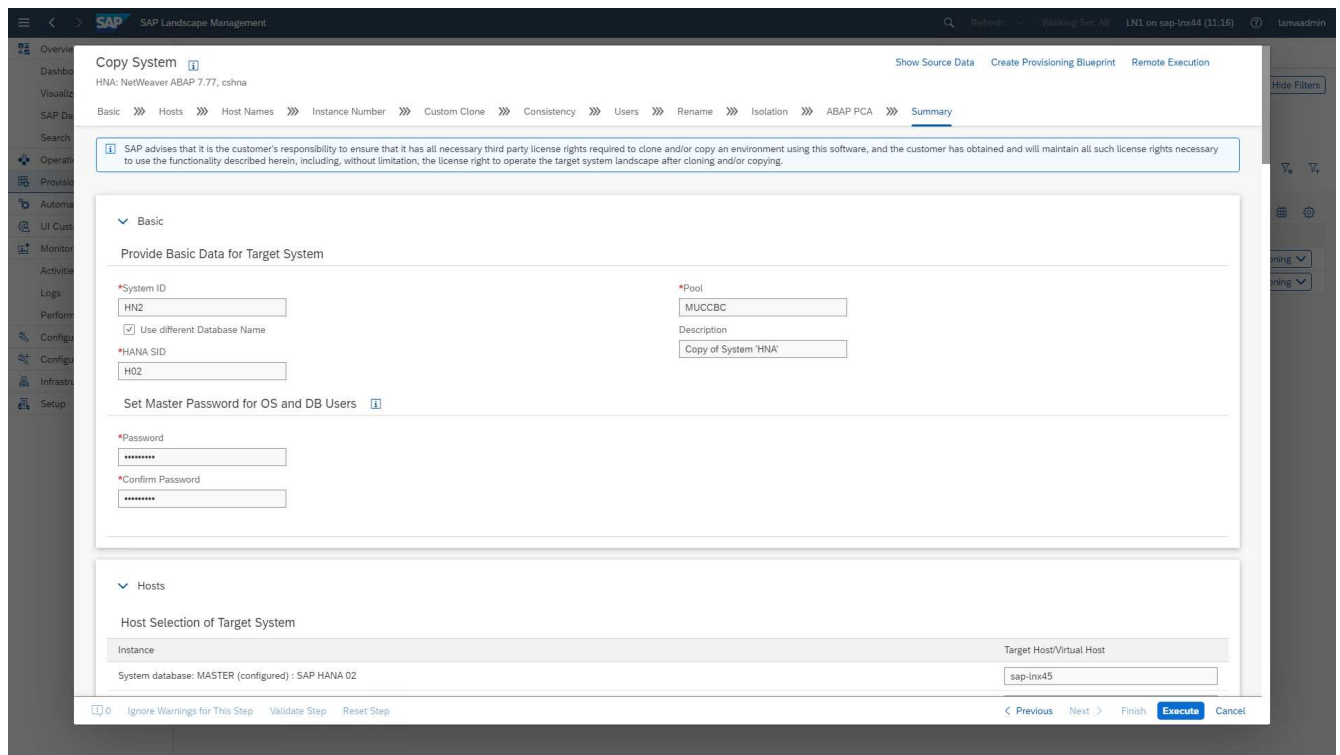


インスタンス番号の変更はテストされていないため、プロバイダスクリプトの変更が必要になる場合があります。

4. 前述のように、* Custom Clone *画面はクローニングワークフローとは異なります。

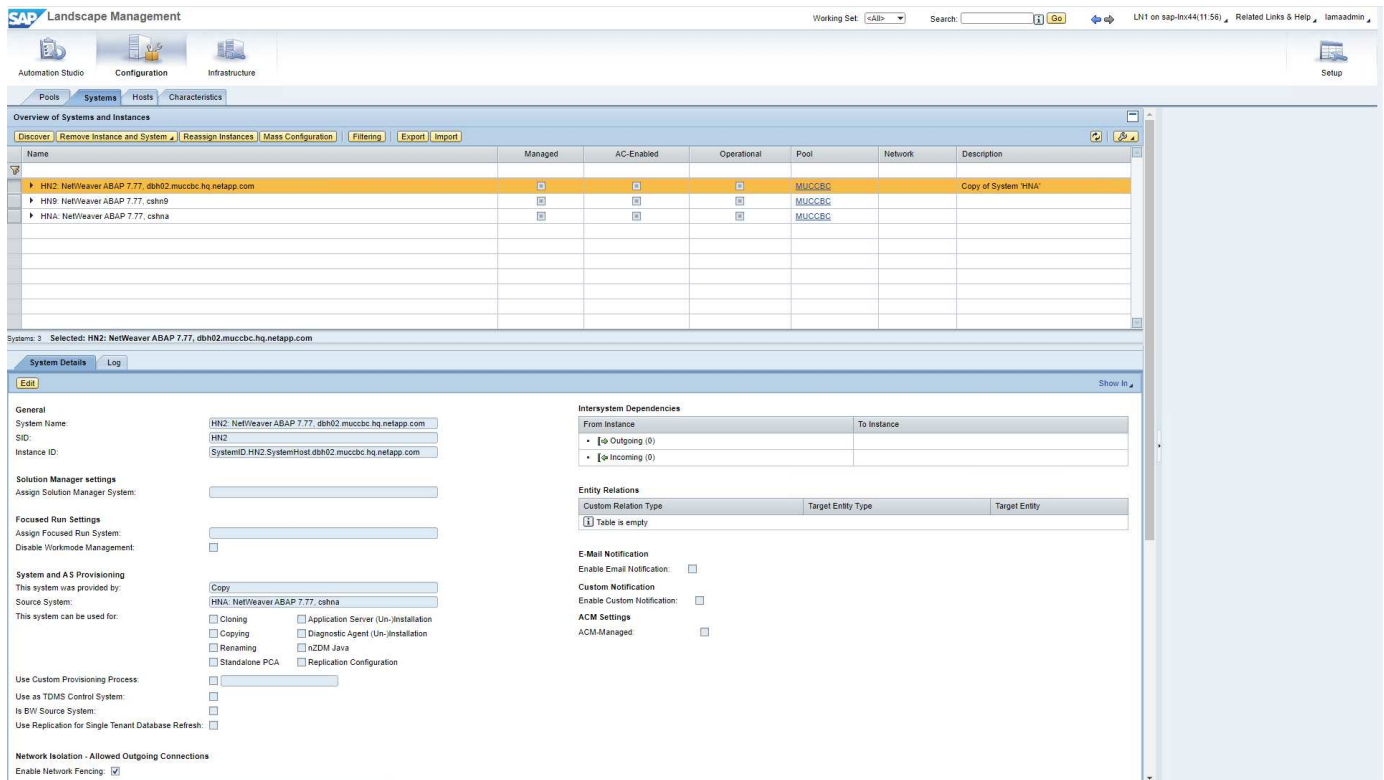


5. すでに説明したように、残りの入力マスクは標準とは異なるので、ここではそれ以上のマスクには入りません。最後の画面に概要が表示され、実行を開始できます。

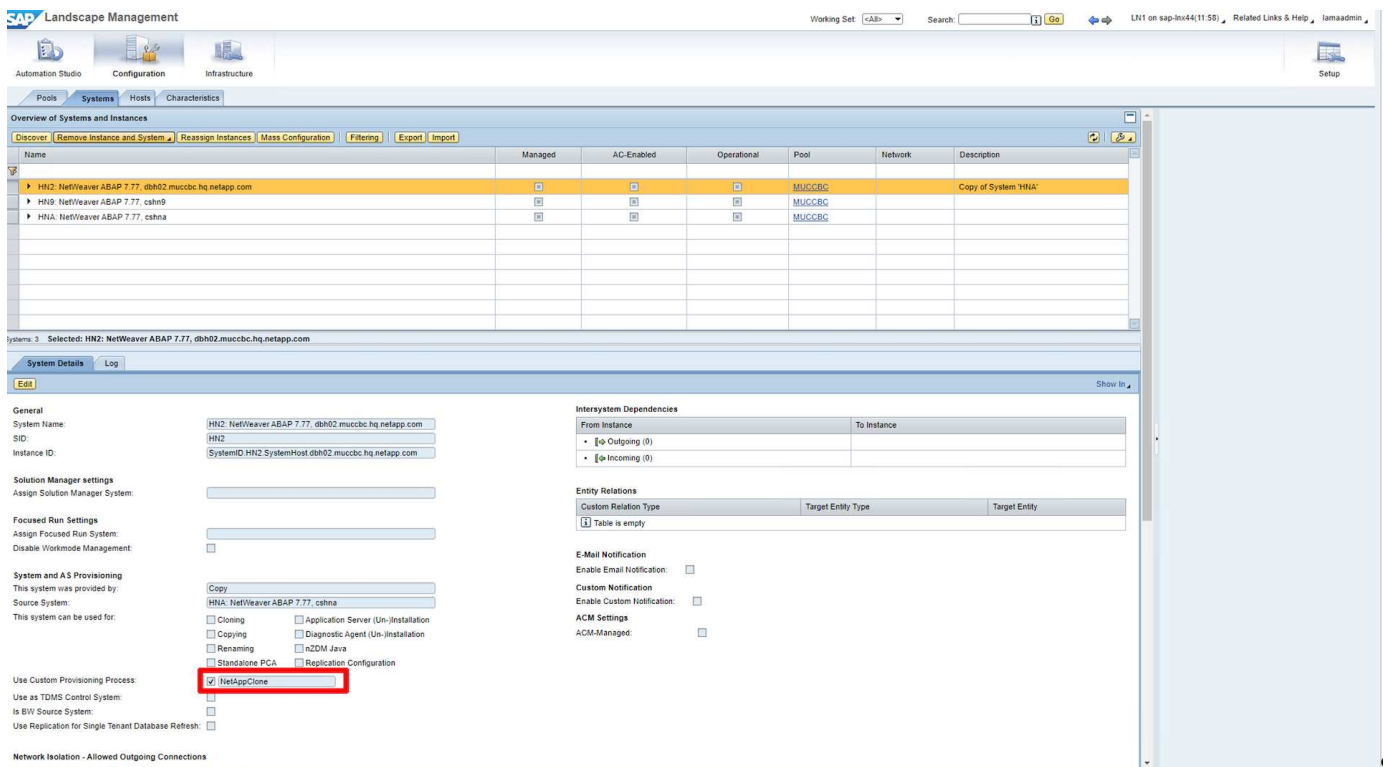


ボタンをクリックします。"]

コピープロセスの完了後、カスタムのクローニングプロセスでターゲットインスタンスが有効になりません。

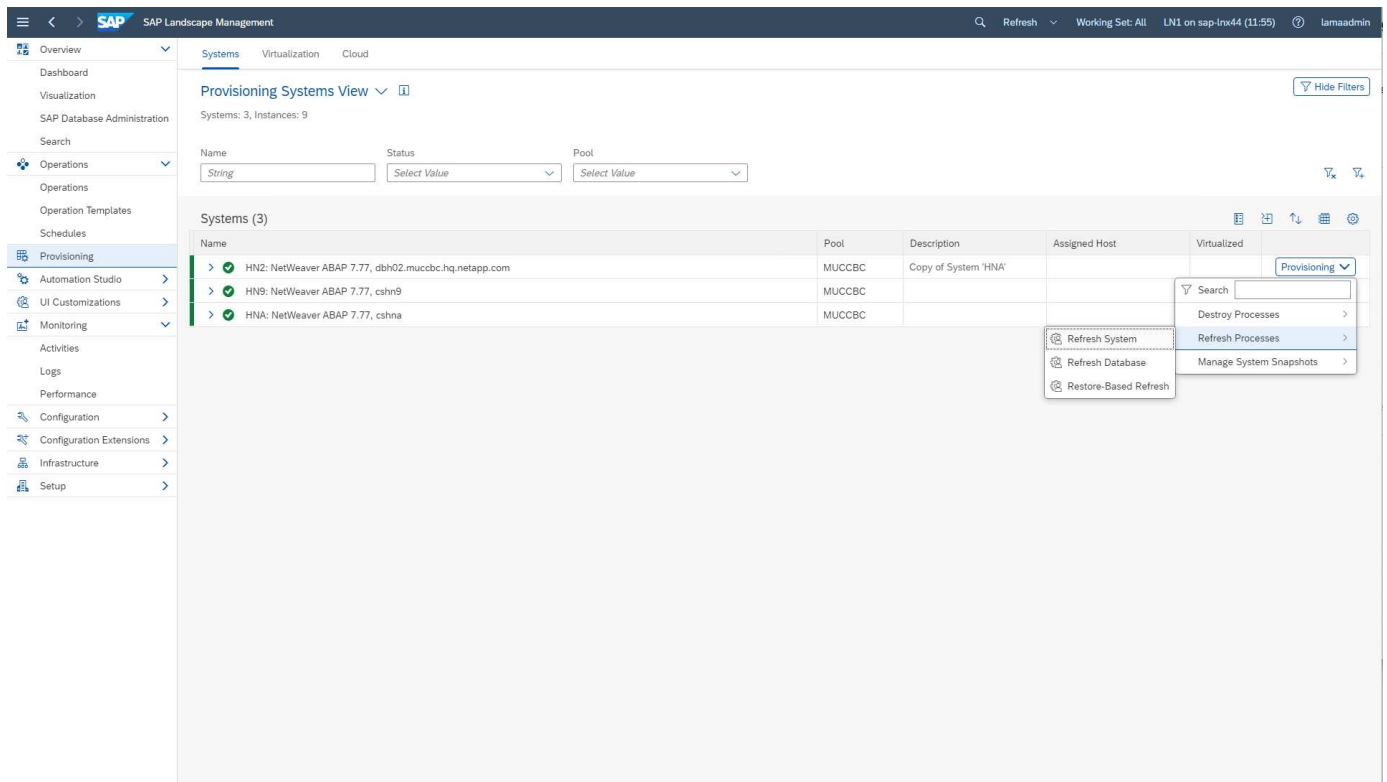


拘束が設定されて実行を妨げるためシステム破壊プロセス中にブックスステップを実行するには'手動で採用する必要があります'

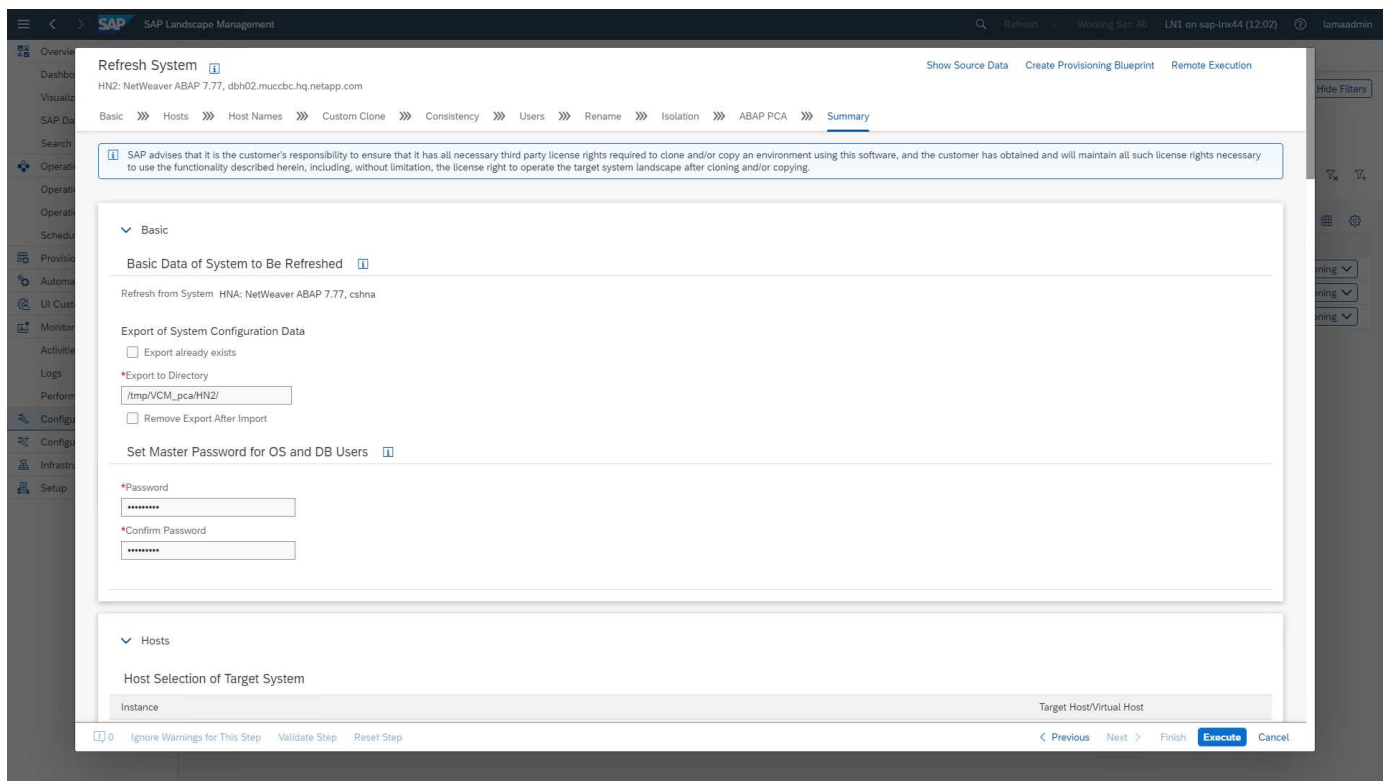


SAP LaMaプロビジョニングワークフロー-システムの更新

次の図は、システムの更新ワークフローで実行する主な手順を示しています。



繰り返しますが、入力画面と標準画面では何も異なりません。ワークフローの実行は概要画面から開始できます。



プロバイダスクリプトの設定とAnsibleのプレイブック

このドキュメントの導入例とワークフローの実行時には、次のプロバイダ構成ファイ

ル、実行スクリプト、およびAnsibleプレイブックを使用します。



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトの最新バージョンは、mailto:ng-sapcc@netapp.com [ng-sapcc@netapp.com]にEメールでリクエストできます。

プロバイダ構成ファイルNetApp_clone.conf

構成ファイルは、の説明に従って作成されます "SAP LaMaドキュメント-「[Configuring SAP Host Agent Registered Scripts](#)」"。この構成ファイルは、SAPホストエージェントがインストールされているAnsibleコントロールノードに配置する必要があります。

設定されているos-user sapuser スクリプトおよび呼び出されたAnsibleプレイブックを実行するための適切な権限が必要です。このスクリプトは共通のスクリプトディレクトリに配置できます。SAP LaMaは、スクリプトを呼び出す際に複数のパラメータを提供します。

カスタムパラメータに加えて、PARAM_ClonePostFix、PROP_ClonePostFix、`PARAM_ClonePostFix` および `PROP_ClonePostFix` に示されているように、他の多くは手渡されることができる "SAP LaMaのドキュメント"。

```
root@sap-jump:~# cat /usr/sap/hostctrl/exe/operations.d/netapp_clone.conf
Name: netapp_clone
Username: sapuser
Description: NetApp Clone for Custom Provisioning
Command: /usr/sap/scripts/netapp_clone.sh
--HookOperationName=${HookOperationName} --SAPSYSTEMNAME=${SAPSYSTEMNAME}
--SAPSYSTEM=${SAPSYSTEM} --MOUNT_XML_PATH=${MOUNT_XML_PATH}
--PARAM_ClonePostFix=${PARAM_ClonePostFix} --PARAM_SnapPostFix=${PARAM
-SnapPostFix} --PROP_ClonePostFix=${PROP_ClonePostFix}
--PROP_SnapPostFix=${PROP_SnapPostFix}
--SAP_LVM_SRC_SID=${SAP_LVM_SRC_SID}
--SAP_LVM_TARGET_SID=${SAP_LVM_TARGET_SID}
ResulConverter: hook
Platform: Unix
```

プロバイダスクリプトnetapp_clone.sh

プロバイダスクリプトはに保存する必要があります /usr/sap/scripts プロバイダの構成ファイルで設定されているとおりです。

変数 (variables)

次の変数はスクリプトでハードコードされており、適宜変更する必要があります。

- PRIMARY_CLUSTER=<hostname of netapp cluster>
- PRIMARY_SVM=<SVM name where source system volumes are stored>

証明書ファイル PRIMARY_KEYFILE=/usr/sap/scripts/ansible/certs/ontap.key および
PRIMARY_CERTFILE=/usr/sap/scripts/ansible/certs/ontap.pem の説明に従って指定する必要があります
あります **"NetApp Ansibleモジュール- ONTAP を準備"**。



それぞれのSAPシステムで異なるクラスタやSVMが必要な場合は、SAP LaMaプロバイダ定義
のパラメータとして追加できます。

機能：インベントリファイルを作成する

Ansibleプレイブックをより動的に実行するために、を使用します inventory. yml その場でファイルが作
成されます。一部の静的値は変数セクションで設定され、一部は実行中に動的に作成されます。

関数：**Ansible**プレイブックを実行する

この関数は、Ansibleプレイブックと動的に作成されるを実行するために使用します inventory.yml ファイル。
プレイブックの命名規則はです netapp_lama_\${HookOperationName}.yml。 の値
\${HookOperationName} LaMaの運用に依存し、LaMaをコマンドラインパラメータとして引き継ぎます。

セクションMain

ここでは、主な実行計画について説明します。変数 \${HookOperationName} LaMaの代替手順の名前が含ま
れ、スクリプトが呼び出されるとLaMaによって提供されます。

- システムクローンとシステムコピーのプロビジョニングワークフローを使用した値：
 - CloneVolumes
 - CloneVolumes
- system destroyワークフローを使用した値：
 - ServiceConfigRemovalサービス構成の削除
- システムの更新ワークフローによる値：
 - ClearMountConfigの場合

HookOperationName=CloneVolumes

この手順では、Ansibleプレイブックが実行され、Snapshotコピーとクローニングの処理が開始されます。ボ
リューム名とマウント設定は、変数で定義されたXMLファイルを使用してSAP LaMaによって引き継がれます
\$MOUNT_XML_PATH。このファイルは、あとで手順で使用するために保存されます
FinalizeCloneVolumes 新しいマウントポイント設定を作成します。ボリューム名がXMLファイルから抽
出され、各ボリュームに対してAnsibleのクローニングプレイブックが実行されます。



この例では、ASインスタンスと中央サービスは同じボリュームを共有します。したがって、ボ
リュームクローニングはSAPインスタンス番号でのみ実行されます (\$SAPSYSTEM) はではありません
01。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName=PostCloneVolumes

このステップでは、ユーザー定義プロパティを指定します ClonePostFix および SnapPostFix また、ター
ゲットシステムのマウントポイント設定が保持されます。

カスタムプロパティは、あとででの運用停止時に入力として使用されます ServiceConfigRemoval または ClearMountConfig フェーズ：システムのプロビジョニングワークフローで指定されたカスタムパラメータの設定を保持するように設計されています。

この例で使用されている値はです ClonePostFix=_clone_20221115 および SnapPostFix=_snap_20221115。

ボリューム HN9_sap`は、動的に作成されるAnsibleファイルに次の値が含まれています。
`datavolumename: HN9_sap、 snapshotpostfix: _snap_20221115`および `clonepostfix: _clone_20221115。

これにより、ボリュームHN9_SAP上のSnapshot名に変換されます HN9_sap_snap_20221115 作成されたボリュームクローンの名前も表示されます HN9_sap_clone_20221115。



カスタムプロパティを使用すると、プロビジョニングプロセスで使用されるパラメータを保持できます。

マウントポイントの設定は、でLaMaに引き継がれたXMLファイルから抽出されます CloneVolume ステップ。。 ClonePostFix がボリューム名に追加され、デフォルトのスクリプト出力でLaMaに返信されます。機能については、を参照してください ["SAPノート1889590"](#)。



この例では、ストレージシステム上のqtreeを、単一のボリュームに異なるデータを配置する一般的な方法として使用します。例： HN9_sap のマウントポイントを保持します /usr/sap/HN9、 /sapmnt/HN9`および `/home/hn9adm。サブディレクトリも同様に機能します。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName = ServiceConfigRemoval

この手順では、ボリュームクローンの削除を実行するAnsibleプレイブックを使用します。

ボリューム名は、マウント構成ファイルとカスタムプロパティを通じてSAP LaMaから引き継がれます ClonePostFix および SnapPostFix は、システムプロビジョニングワークフローで最初に指定されたパラメータの値を渡すために使用します（の注を参照） HookOperationName = PostCloneVolumes）。

ボリューム名がXMLファイルから抽出され、各ボリュームに対してAnsibleのクローニングプレイブックが実行されます。



この例では、ASインスタンスと中央サービスは同じボリュームを共有します。そのため、ボリュームの削除はSAPインスタンス番号がの場合にのみ実行されます (\$SAPSYSTEM) はではありません 01。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName=ClearMountConfig

この手順では、システムの更新ワークフロー中にボリュームクローンを削除するAnsibleプレイブックを実行します。

ボリューム名は、マウント構成ファイルとカスタムプロパティを通じてSAP LaMaから引き継がれます ClonePostFix および SnapPostFix は、システムプロビジョニングワークフローで最初に指定されたパラメータの値を渡すために使用します。

ボリューム名がXMLファイルから抽出され、各ボリュームに対してAnsibleのクローニングプレイブックが実

行されます。



この例では、ASインスタンスと中央サービスは同じボリュームを共有します。そのため、ボリュームの削除はSAPインスタンス番号がの場合にのみ実行されます (\$SAPSYSTEM) はではありません 01。これは他の環境とは異なる場合があるため、変更する必要があります。

```
root@sap-jump:~# cat /usr/sap/scripts/netapp_clone.sh
#!/bin/bash
#Section - Variables
#####
VERSION="Version 0.9"
#Path for ansible play-books
ANSIBLE_PATH=/usr/sap/scripts/ansible
#Values for Ansible Inventory File
PRIMARY_CLUSTER=grenada
PRIMARY_SVM=svm-sap01
PRIMARY_KEYFILE=/usr/sap/scripts/ansible/certs/ontap.key
PRIMARY_CERTFILE=/usr/sap/scripts/ansible/certs/ontap.pem
#Default Variable if PARAM ClonePostFix / SnapPostFix is not maintained in
LaMa
DefaultPostFix=_clone_1
#TMP Files - used during execution
YAML_TMP=/tmp/inventory_ansible_clone_tmp_$.yml
TMPFILE=/tmp/tmpfile.$$
MY_NAME="`basename $0`"
BASE_SCRIPT_DIR="`dirname $0`"
#Sendig Script Version and run options to LaMa Log
echo "[DEBUG]: Running Script $MY_NAME $VERSION"
echo "[DEBUG]: $MY_NAME $@"
#Command declared in the netapp_clone.conf Provider definition
#Command: /usr/sap/scripts/netapp_clone.sh
--HookOperationName=${HookOperationName} --SAPSYSTEMNAME=${SAPSYSTEMNAME}
--SAPSYSTEM=${SAPSYSTEM} --MOUNT_XML_PATH=${MOUNT_XML_PATH}
--PARAM_ClonePostFix=${PARAM_ClonePostFix} --PARAM_SnapPostFix=${PARAM
-SnapPostFix} --PROP_ClonePostFix=${PROP_ClonePostFix}
--PROP_SnapPostFix=${PROP_SnapPostFix}
--SAP_LVM_SRC_SID=${SAP_LVM_SRC_SID}
--SAP_LVM_TARGET_SID=${SAP_LVM_TARGET_SID}
#Reading Input Variables hand over by LaMa
for i in "$@"
do
case $i in
--HookOperationName=*)
HookOperationName="${i#*=}";shift;;
--SAPSYSTEMNAME=*)
SAPSYSTEMNAME="${i#*=}";shift;;
```

```

--SAPSYSTEM=*)
SAPSYSTEM="${i#*=}";shift;;
--MOUNT_XML_PATH=*)
MOUNT_XML_PATH="${i#*=}";shift;;
--PARAM_ClonePostFix=*)
PARAM_ClonePostFix="${i#*=}";shift;;
--PARAM_SnapPostFix=*)
PARAM_SnapPostFix="${i#*=}";shift;;
--PROP_ClonePostFix=*)
PROP_ClonePostFix="${i#*=}";shift;;
--PROP_SnapPostFix=*)
PROP_SnapPostFix="${i#*=}";shift;;
--SAP_LVM_SRC_SID=*)
SAP_LVM_SRC_SID="${i#*=}";shift;;
--SAP_LVM_TARGET_SID=*)
SAP_LVM_TARGET_SID="${i#*=}";shift;;
*)
# unknown option
;;
esac
done
#If Parameters not provided by the User - defaulting to DefaultPostFix
if [ -z $PARAM_ClonePostFix ]; then PARAM_ClonePostFix=$DefaultPostFix;fi
if [ -z $PARAM_SnapPostFix ]; then PARAM_SnapPostFix=$DefaultPostFix;fi
#Section - Functions
#####
#Function Create (Inventory) YML File
#####
create_yaml_file()
{
echo "ontapservers:">$YAML_TMP
echo " hosts:">>$YAML_TMP
echo "   ${PRIMARY_CLUSTER}:">>$YAML_TMP
echo "   ansible_host: '$PRIMARY_CLUSTER'>>$YAML_TMP
echo "   keyfile: '$PRIMARY_KEYFILE'>>$YAML_TMP
echo "   certfile: '$PRIMARY_CERTFILE'>>$YAML_TMP
echo "   svmname: '$PRIMARY_SVM'>>$YAML_TMP
echo "   datavolumename: '$datavolumename'>>$YAML_TMP
echo "   snapshotpostfix: '$snapshotpostfix'>>$YAML_TMP
echo "   clonepostfix: '$clonepostfix'>>$YAML_TMP
}
#Function run ansible-playbook
#####
run_ansible_playbook()
{
echo "[DEBUG]: Running ansible playbook

```

```

netapp_lama_${HookOperationName}.yaml on Volume $datavolumename"
ansible-playbook -i $YAML_TMP
$ANSIBLE_PATH/netapp_lama_${HookOperationName}.yaml
}
#Section - Main
#####
#HookOperationName - CloneVolumes
#####
if [ $HookOperationName = CloneVolumes ] ;then
#save mount xml for later usage - used in Section FinalizeCloneVolumes to
generate the mountpoints
echo "[DEBUG]: saving mount config...."
cp $MOUNT_XML_PATH /tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
#Instance 00 + 01 share the same volumes - clone needs to be done once
if [ $SAPSYSTEM != 01 ]; then
#generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtree"
xmlFile=/tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
if [ -e $TMPFILE ];then rm $TMPFILE;fi
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
    xmllint --xpath "/mountconfig/mount[$i]/exportpath/text()" $xmlFile
|awk -F"/" '{print $2}' >>$TMPFILE
i=$((i + 1))
done
DATAVOLUMES=`cat $TMPFILE |sort -u`
#Create yaml file and rund playbook for each volume
for I in $DATAVOLUMES; do
datavolumename="$I"
snapshotpostfix="$PARAM_SnapPostFix"
clonepostfix="$PARAM_ClonePostFix"
create_yaml_file
run_ansible_playbook
done
else
echo "[DEBUG]: Doing nothing .... Volume cloned in different Task"
fi
fi
#HookOperationName - PostCloneVolumes
#####
if [ $HookOperationName = PostCloneVolumes ] ;then
#Reporting Properties back to LaMa Config for Cloned System
echo "[RESULT]:Property:ClonePostFix=$PARAM_ClonePostFix"
echo "[RESULT]:Property:SnapPostFix=$PARAM_SnapPostFix"

```

```

#Create MountPoint Config for Cloned Instances and report back to LaMa
according to SAP Note: https://launchpad.support.sap.com/#/notes/1889590
echo "MountDataBegin"
echo '<?xml version="1.0" encoding="UTF-8"?>'
echo "<mountconfig>"
xmlFile=/tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
MOUNTPPOINT=`xmllint --xpath "/mountconfig/mount[$i]/mountpoint/text()"
$xmlFile`;
        EXPORTPATH=`xmllint --xpath
"/mountconfig/mount[$i]/exportpath/text()" $xmlFile`;
        OPTIONS=`xmllint --xpath "/mountconfig/mount[$i]/options/text()"
$xmlFile`;
#Adopt Exportpath and add Clonepostfix - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
TMPFIELD1=`echo $EXPORTPATH|awk -F"/" '{print $1}'`
TMPFIELD2=`echo $EXPORTPATH|awk -F"/" '{print $2}'`
TMPFIELD3=`echo $EXPORTPATH|awk -F"/" '{print $3}'`
EXPORTPATH=$TMPFIELD1":/`${TMPFIELD2}$PARAM_ClonePostFix"/`${TMPFIELD3
echo -e '\t<mount fstype="nfs" storagetype="NETFS">'
echo -e "\t\t<mountpoint>${MOUNTPPOINT}</mountpoint>"
echo -e "\t\t<exportpath>${EXPORTPATH}</exportpath>"
echo -e "\t\t<options>${OPTIONS}</options>"
echo -e "\t</mount>"
i=$((i + 1))
done
echo "</mountconfig>"
echo "MountDataEnd"
#Finished MountPoint Config
#Cleanup Temporary Files
rm $xmlFile
fi
#HookOperationName - ServiceConfigRemoval
#####
if [ $HookOperationName = ServiceConfigRemoval ] ;then
#Assure that Properties ClonePostFix and SnapPostfix has been configured
through the provisioning process
if [ -z $PROP_ClonePostFix ]; then echo "[ERROR]: Propertiy ClonePostFix
is not handed over - please investigate";exit 5;fi
if [ -z $PROP_SnapPostFix ]; then echo "[ERROR]: Propertiy SnapPostFix is
not handed over - please investigate";exit 5;fi
#Instance 00 + 01 share the same volumes - clone delete needs to be done
once

```

```

if [ $SAPSYSTEM != 01 ]; then
#generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
xmlFile=$MOUNT_XML_PATH
if [ -e $TMPFILE ];then rm $TMPFILE;fi
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
    xmllint --xpath "/mountconfig/mount[$i]/exportpath/text()" $xmlFile
|awk -F"/" '{print $2}' >>$TMPFILE
i=$((i + 1))
done
DATAVOLUMES=`cat $TMPFILE |sort -u| awk -F $PROP_ClonePostFix '{ print $1
}'`
#Create yml file and rund playbook for each volume
for I in $DATAVOLUMES; do
datavolumename="$I"
snapshotpostfix="$PROP_SnapPostFix"
clonepostfix="$PROP_ClonePostFix"
create_yml_file
run_ansible_playbook
done
else
echo "[DEBUG]: Doing nothing .... Volume deleted in different Task"
fi
#Cleanup Temporary Files
rm $xmlFile
fi
#HookOperationName - ClearMountConfig
#####
if [ $HookOperationName = ClearMountConfig ] ;then
    #Assure that Properties ClonePostFix and SnapPostfix has been
configured through the provisioning process
    if [ -z $PROP_ClonePostFix ]; then echo "[ERROR]: Propertiy
ClonePostFix is not handed over - please investigate";exit 5;fi
    if [ -z $PROP_SnapPostFix ]; then echo "[ERROR]: Propertiy
SnapPostFix is not handed over - please investigate";exit 5;fi
    #Instance 00 + 01 share the same volumes - clone delete needs to
be done once
    if [ $SAPSYSTEM != 01 ]; then
        #generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
        xmlFile=$MOUNT_XML_PATH
        if [ -e $TMPFILE ];then rm $TMPFILE;fi
        numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile

```

```

| grep "total: " | awk '{ print $2 }'`
        i=1
        while [ $i -le $numMounts ]; do
            xmllint --xpath
"/mountconfig/mount[$i]/exportpath/text()" $xmlFile |awk -F"/" '{print
$2}' >>$TMPFILE
            i=$((i + 1))
        done
        DATAVOLUMES=`cat $TMPFILE |sort -u| awk -F
$PROP_ClonePostFix '{ print $1 }'`
        #Create yaml file and rund playbook for each volume
        for I in $DATAVOLUMES; do
            datavolumename="$I"
            snapshotpostfix="$PROP_SnapPostFix"
            clonepostfix="$PROP_ClonePostFix"
            create_yaml_file
            run_ansible_playbook
        done
    else
        echo "[DEBUG]: Doing nothing .... Volume deleted in
different Task"
    fi
    #Cleanup Temporary Files
    rm $xmlFile
fi
#Cleanup
#####
#Cleanup Temporary Files
if [ -e $TMPFILE ];then rm $TMPFILE;fi
if [ -e $YAML_TMP ];then rm $YAML_TMP;fi
exit 0

```

Ansible Playbook : NetApp_LaMa_CloneVolume.yml

LaMaシステムのクローニングワークフローのCloneVolumesの手順で実行されるPlaybookは、を組み合わせたものです create_snapshot.yml および create_clone.yml (を参照) ["NetApp Ansibleモジュール-YAMLファイル"](#))。このプレイブックは、セカンダリからのクローニング処理やクローンスプリット処理など、他のユースケースにも簡単に対応できます。

```

root@sap-jump:~# cat /usr/sap/scripts/ansible/netapp_lama_CloneVolumes.yml
---
- hosts: ontapservers
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_CloneVolumes
  tasks:
    - name: Create SnapShot
      na_ontap_snapshot:
        state: present
        snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        use_rest: always
        volume: "{{ datavolumename }}"
        vserver: "{{ svmname }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
    - name: Clone Volume
      na_ontap_volume_clone:
        state: present
        name: "{{ datavolumename }}{{ clonepostfix }}"
        use_rest: always
        vserver: "{{ svmname }}"
        junction_path: '/{{ datavolumename }}{{ clonepostfix }}'
        parent_volume: "{{ datavolumename }}"
        parent_snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false

```

Ansible Playbook : NetApp_LaMa _ServiceConfigRemoval.yml

実行されるプレイブック ServiceConfigRemoval LaMaシステムの破棄ワークフローのフェーズは、のフェーズです delete_clone.yml および delete_snapshot.yml （を参照）["NetApp Ansibleモジュール-YAMLファイル"](#)）。の実行ステップに合わせて調整する必要があります netapp_lama_CloneVolumes Playbook :


```

root@sap-jump:~# cat
/usr/sap/scripts/ansible/netapp_lama_ServiceConfigRemoval.yml
---
- hosts: ontapservers
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_ServiceConfigRemoval
  tasks:
    - name: Delete Clone
      na_ontap_volume:
        state: absent
        name: "{{ datavolumename }}{{ clonepostfix }}"
        use_rest: always
        vservers: "{{ svmname }}"
        wait_for_completion: True
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
    - name: Delete SnapShot
      na_ontap_snapshot:
        state: absent
        snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        use_rest: always
        volume: "{{ datavolumename }}"
        vservers: "{{ svmname }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
root@sap-jump:~#

```

Ansible Playbook : NetApp_LaMa _ClearMountConfig.yml

プレイブックは、の実行時に指定します netapp_lama_ClearMountConfig LaMaシステムの更新ワークフローのフェーズは、のフェーズです delete_clone.yml および delete_snapshot.yml （を参照）["NetApp Ansibleモジュール- YAMLファイル"](#)）。の実行ステップに合わせて調整する必要があります

netapp_lama_CloneVolumes Playbook :

```

root@sap-jump:~# cat
/usr/sap/scripts/ansible/netapp_lama_ServiceConfigRemoval.yml
---
- hosts: ontapservers
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_ServiceConfigRemoval
  tasks:
    - name: Delete Clone
      na_ontap_volume:
        state: absent
        name: "{{ datavolumename }}{{ clonepostfix }}"
        use_rest: always
        vserver: "{{ svmname }}"
        wait_for_completion: True
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
    - name: Delete SnapShot
      na_ontap_snapshot:
        state: absent
        snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        use_rest: always
        volume: "{{ datavolumename }}"
        vserver: "{{ svmname }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
root@sap-jump:~#

```

Ansibleのinventory.ymlの例

このインベントリファイルは、ワークフローの実行時に動的に作成されます。このファイルは、説明のためにのみここに表示されています。

```
ontapservers:
  hosts:
    grenada:
      ansible_host: "grenada"
      keyfile: "/usr/sap/scripts/ansible/certs/ontap.key"
      certfile: "/usr/sap/scripts/ansible/certs/ontap.pem"
      svmname: "svm-sap01"
      datavolumename: "HN9_sap"
      snapshotpostfix: " _snap_20221115"
      clonepostfix: " _clone_20221115"
```

まとめ

Ansibleなどの最新の自動化フレームワークがSAP LaMaプロビジョニングワークフローに統合されているため、柔軟な解決策を利用して、標準的なインフラや複雑なインフラ要件に対応できます。

追加情報の参照先

このドキュメントに記載されている情報の詳細については、以下のドキュメントや Web サイトを参照してください。

- NetAppネームスペース内のコレクション

["https://docs.ansible.com/ansible/latest/collections/netapp/index.html"](https://docs.ansible.com/ansible/latest/collections/netapp/index.html)

- AnsibleとAnsibleのPlaybookのサンプルに関するドキュメント

["https://github.com/sap-linuxlab/demo.netapp_ontap"](https://github.com/sap-linuxlab/demo.netapp_ontap)

- Ansibleとネットアップの統合

["https://www.ansible.com/integrations/infrastructure/netapp"](https://www.ansible.com/integrations/infrastructure/netapp)

- SAP LaMaとAnsibleの統合に関するブログ

["https://blogs.sap.com/2020/06/08/outgoing-api-calls-from-sap-landscape-management-lama-with-automation-studio/"](https://blogs.sap.com/2020/06/08/outgoing-api-calls-from-sap-landscape-management-lama-with-automation-studio/)

- SAP Landscape Management 3.0, Enterprise Edition Documentation

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/4df88a8f418c5059e1000000a42189c.html#loio4df88a8f418c5059e1000000a42189c"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/4df88a8f418c5059e1000000a42189c.html#loio4df88a8f418c5059e1000000a42189c)

- SAP LaMaのドキュメント–プロバイダの定義

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/bf6b3e43340a4cbcb0c0f3089715c068.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/bf6b3e43340a4cbcb0c0f3089715c068.html)

- SAP LaMaドキュメント-カスタムフック

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/139eca2f925e48738a20dbf0b56674c5.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/139eca2f925e48738a20dbf0b56674c5.html)

- SAP LaMaドキュメント-「Configuring SAP Host Agent Registered Scripts

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/250dfc5eef4047a38bab466c295d3a49.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/250dfc5eef4047a38bab466c295d3a49.html)

- SAP LaMaドキュメント-カスタム運用とカスタムフックのパラメータ

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/0148e495174943de8c1c3ee1b7c9cc65.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/0148e495174943de8c1c3ee1b7c9cc65.html)

- SAP LaMaドキュメント-アダプティブデザイン

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/737a99e86f8743bdb8d1f6cf4b862c79.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/737a99e86f8743bdb8d1f6cf4b862c79.html)

- ネットアップの製品マニュアル

["https://www.netapp.com/support-and-training/documentation/"](https://www.netapp.com/support-and-training/documentation/)

バージョン履歴

バージョン	日付	ドキュメントのバージョン履歴
バージョン 1.0 以降	2023年1月	初版リリース

著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。